2006-12

Development and implementation of new control law for vision based target tracking system onboard small Unmanned Aerial Vehicles

Tay, Boon Chong.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

DEVELOPMENT AND IMPLEMENTATION OF NEW CONTROL LAW FOR VISION BASED TARGET TRACKING SYSTEM ONBOARD SMALL UNMANNED AERIAL VEHICLES

by

Tay Boon Chong

December 2006

Thesis Advisor:	Issac I. Kaminer
Co- Advisor:	Vladimir N. Dobrokhodov

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 2006	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE : Development and Implementation of a New Control Law for the Vision Based Target Tracking System Onboard Small Unmanned Aerial Vehicles	5. FUNDING NUMBERS
6. AUTHOR(S) Tay Boon Chong	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)

A new control law is being developed and implemented for the Vision Based Target Tracking (VBTT) system onboard a small unmanned aerial vehicle (SUAV). The new control law allows for coordinated SUAV guidance and vision-based target tracking of stationary and moving targets in the presence of atmospheric disturbances and measurements noise. The new control law is tested for its performance and stability in both the theoretical 6DOF simulation and the Hardware-in-the-Loop (HIL) simulation. Principal results show that realistic measures of performance of the control law are continuous and exhibit predictable degradation of performance with increase of target speed. The results are encouraging and comparable among theoretical predictions, actual hardware simulation results, and initial flight testing. The control law development, implementation, and trial processes and procedures are also examined and categorically documented in this thesis as future reference on the subject development, as well as for better knowledge retention, continuation and proliferation of the VBTT system.

14. SUBJECT TERMS Unmanned Aerial Vehicle, UAV, Autonomous Guidance, Target Tracking, Control, Vision Based Target Tracking, Simulink, Hardware in the loop, Piccolo, xPC Target, PC-104	15. NUMBER OF PAGES 109
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

DEVELOPMENT AND IMPLEMENTATION OF A NEW CONTROL LAW FOR THE VISION BASED TARGET TRACKING SYSTEM ONBOARD SMALL UNMANNED AERIAL VEHICLES

Boon Chong, Tay
Lieutenant Colonel, Republic of Singapore Air Force
B.Eng. (Aeronautical Engineering), London Imperial College, 1992

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING SCIENCE
(MECHANICAL ENGINEERING)

from the

NAVAL POSTGRADUATE SCHOOL
December 2006

Author: Tay Boon Chong

Approved by: Professor Issac I. Kaminer
Thesis Advisor

Professor Vladimir N. Dobrokhodov
Co-Advisor

Professor Anthony J Healey
Chairman, Department of Mechanical and Astronautical Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

A new control law is being developed and implemented for the Vision Based Target Tracking (VBTT) system onboard a small unmanned aerial vehicle (SUAV). The new control law allows for coordinated SUAV guidance and vision-based target tracking of stationary and moving targets in the presence of atmospheric disturbances and measurements noise. The new control law is tested for its performance and stability in both the theoretical 6DOF simulation and the Hardware-in-the-Loop (HIL) simulation. Principal results show that realistic measures of performance of the control law are continuous and exhibit predictable degradation of performance with increase of target speed. The results are encouraging and comparable among theoretical predictions, actual hardware simulation results, and initial flight testing. The control law development, implementation, and trial processes and procedures are also examined and categorically documented in this thesis as future reference on the subject development, as well as for better knowledge retention, continuation and proliferation of the VBTT system.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

The author would like to thank Professor Issac Kaminer for his consents, for providing the opportunity to work on this interesting project, and for all assistance he rendered along the way.

The author would also like to thank Professor Vladimir Dobrokhodov for all his valuable help and patience in explaining the difficult concepts. Professor Dobrokhodov also provided invaluable guidance on the development and implementation of the modifications to the VBTT control system.

The author would also like to thank Professor Kevin Jones for all help he rendered during the course of project implementation.

Finally, the author would like to acknowledge the important remote support and encouragement he received from his wife and three adorable children back in Singapore. In the course of pursuing a Masters degree in NPS alone here in the USA, they have kept him rightly focused and motivated on the accomplishment of this thesis project.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

The Modern Unmanned Aerial Vehicle is an autonomous surveillance platform that has garnered prominent and important roles in today's battlefield. At present, it is favorably situated to expand and extend its prominences and successes, in advancing its operational and technological influences for the research and development of the future network centric warfare concepts.

Central to the realization of a fully unmanned autonomously synchronous surveillance goal is the culmination and amalgamation of several maturing technologies. These technologies encompass areas of a low cost, light weight unmanned aerial vehicle, high speed wireless network communication technology, and real-time simulation software for rapidly deployed hardware reconfiguration of advanced guidance and control algorithms.

B. PROBLEM STATEMENT

In the modern day's battlefield and the future network centric battle-space environment, building and maintaining a dynamic information and intelligence network architecture is a crucial and fundamental battlefield task. The integrated information overlay that is constructed from the collected data will enable military missions and tasks to jointly utilize and leverage the real-time information and intelligence in order to pursue illusive enemies and fleeting targets in the most efficient and effective manner.

To aid in the information management process, fully autonomous unmanned aerial sensor platform and surveillance automation will be indispensable to alleviating the problem of human constraints. The necessity of controlling multiple airborne platforms will only complicate this task. A control law algorithm that automatically couples the dual objectives of maneuvering the unmanned aerial sensor platform and its surveillance sensor will vastly aid the operator, enabling him to work with more spare cognitive capacity. Therefore, the operator can better manage a great deal of critical information, and can timely process key intelligence. The operator is protected from potential

saturation while performing multiple tasks, such as the management of the platform flight dynamics, airspace de-confliction and safety, and the onboard sensor control.

C. OVERVIEW OF TACTICAL NETWORK TOPOLOGY (TNT) EXPERIMENT AND ITS UAV SEGMENT

The current research is an integral part of the Tactical Network Topology (TNT) field experimentation program, a cooperative effort between the Naval Postgraduate School (NPS), USSOCOM, and its component commands. The program—conducted quarterly at the Center for Interdisciplinary Remotely-Piloted Aircraft Studies (CIRPAS) facility located at McMillan air field in Camp Roberts, CA—is a continuation of the Surveillance and Target Acquisition (STAN) program, and focuses on the exploration of network-based warfare capabilities and their integration into current real-world situations.

This thesis presents the development and testing of a VBTT system. The system controls a UAV and a gimbaled camera to keep the operator-selected target in the center of the video image and provide an estimate of target GPS position. The target can be stationary or moving.

The VBTT system includes a Senior Telemaster SUAV that was modified to carry a two-axis gimbaled camera, which acquires video and sends the information to the automatic target tracking (ATT) computer in real time. During a mission, the operator of the ATT computer may identify the target of interest. The target appears inside of a small rectangular polygon and is tracked by engaging the “Track mode”. The position of the target is identified by Cartesian coordinates in a camera frame. This passive information is processed by the control algorithm that sends commands to the SUAV and to the gimbaled camera to keep the target in the center of the video frame.

Figure 1: Modified Telemaster UAV

In order to make the VBTT, three major components were developed and integrated into one system. The first one includes a vision-based target tracking capability that uses imagery provided by a gimbaled camera. Development of this component involved the design of a miniaturized gimbaled camera and a controller, and integration of the automated motion tracking software by PerceptiVU, Inc.

The second component of the system includes the control law for the SUAV. The control algorithm was designed to solve two principal tasks. First, it had to navigate the SUAV around the target while keeping the target in the camera frame. Second, it had to reduce the range estimation errors, because the accuracy of the range estimation depends on the translating motion of the camera. The estimation error is minimized when the target moves parallel to the camera image plane.

In general, in order to estimate the target position efficiently, the target must exhibit translational motion in the camera frame tracked from the SUAV. The differences in the translation motions of the target in the camera frame for frontal and circular approaches are presented in Figure 2:

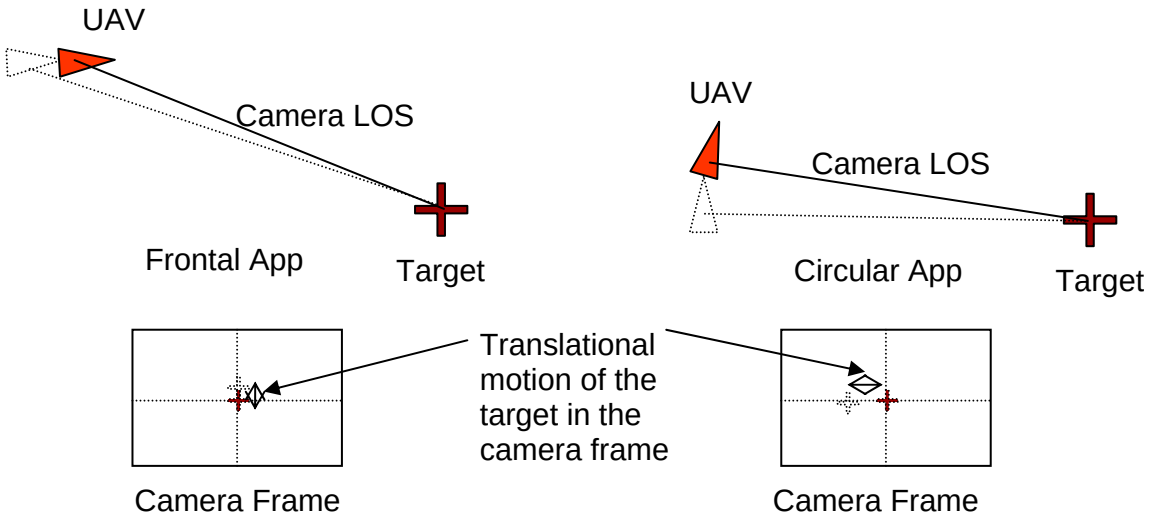


Figure 2: Translation Motions of Target in Camera Frame

In a frontal approach situation, where the SUAV flight path and its camera LOS is pointing forward and towards the target, the translational motion of the target in the camera frame is minimal at large distances, and consequently a large sampling time interval must be used in order to provide low dilution of precision (DOP) of the target position. An alternate strategy is to circle the SUAV around the target with the camera mid line pointing 90 degrees away from the SUAV forward velocity. This approach, which is traditionally used in triangulation, ensures maximum translational motion of the target in the camera frame, and thus the sampling time interval can be significantly smaller for a continuous position estimate of the target. Although triangulation is not used in current projects for target motion estimation, the idea of translational motion and its effect onto position estimation is still valid and is used for control law development.

The control concept is depicted in the Figure 3, below. It illustrates a shape of an orbit above the center of the target, while the SUAV is autonomously guided to accomplish the task of target tracking. The VBTT guidance algorithm controls the ground speed vector V_g of the UAV in such a way as to make it continuously perpendicular to the Line of sight (LOS) [Ref1, Ref 2 and Ref 7]. This control strategy guarantees a maximum of the translational motion mentioned above.

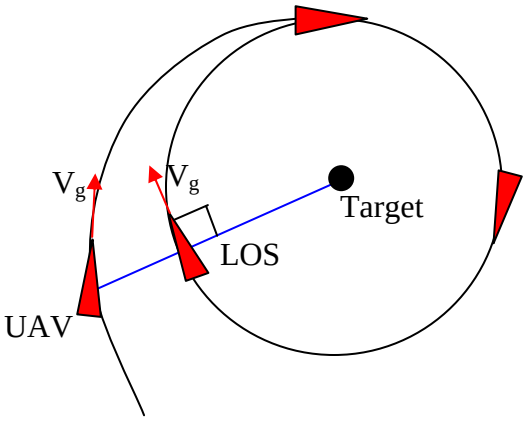


Figure 3: Conceptual Depiction of VBTT Guidance Algorithm

The third component consisted of two filters for target position estimation (not considered in this thesis).

D. THESIS OBJECTIVES

The objective of this thesis is twofold. The first objective is to develop and implement a new control law algorithm for the VBTT system for SUAV, independent of target speed. The next logical goal of this projects concerns Hardware-In-The-Loop (HIL) lab experimentation and following flight test experiments. The documentation of the research, development, simulation and hardware implementation of the VBTT system is also examined and presented categorically for the purpose of knowledge retention, continuation, and proliferation

THIS PAGE INTENTIONALLY LEFT BLANK

II. CONTROL LAW DEVELOPMENT

A. COORDINATE SYSTEMS

Multiple coordinate frames are often used to define the motions or behaviors of an object in complex dynamic systems. This preference and practice stems from the fact that it is often easier to define the motion of an object in one convenient frame (usually a coordinate frame that is related directly to the object's own "natural" axes system) rather than any arbitrary reference frame . Hence the translational and rotational kinematics states; i.e. the positions, velocities, and accelerations of the objects can be fully described in any frame as long as the transformation relationships between the coordinate frames are known. The following coordinate systems were used in the development of the control law in this thesis. (All information on coordinate systems is taken from Ref 2, Ref 3, Ref 5 and Ref 6.)

1. Navigation Inertia Coordinate Frame (I – Frame)

This local level frame assumed a flat earth model in the vicinity of the reference navigation point; the $X_n - Y_n$ axes lie in a plane tangent to the reference point origin on the earth ellipsoid surface, and the Z_n axis lie perpendicular to that ellipsoid surface. It is assumed that the X_n axis points north, the Y_n axis east, and the Z_n axis down. This is a north-east-down (NED) coordinate system, also known as Local Tangent Plane (LTP).

2. SUAV Body Coordinate Frame (B – Frame)

A convenient coordinate system for developing the equations of motion of the SUAV is a right-hand orthogonal system whose origin centers at the aircraft's center of gravity. Conventionally, the x-axis points forward along the longitudinal axis of the aircraft, the y-axis points outwards towards the right wing, and the z-axis is in the downward direction.

3. Gimbal Platform Coordinate Frame (G – Frame)

The gimbal coordinate frame is a right-hand orthogonal coordinate system whose origin is the location of the camera mount. The x-axis of the gimbal frame points forward along the longitudinal axis of the gimbal platform, the y-axis points outward toward the right hand side, and the z-axis downward from the gimbal platform.

4. Camera Coordinate Frame (C – Frame)

The camera coordinate frame is a right-hand orthogonal coordinate system whose origin is located at the focal point of the camera. The x-axis points forward along the longitudinal axis of the camera, the y-axis points outwards toward the right hand side, and the z-axis points downward from the origin.

5. Image Plane Coordinate Frame (P – Frame)

The image plane reference frame is the coordinate system used to describe the location of the target in the image plane. It is a 2-dimensional coordinate system with the u-axis aligned with the y-axis of the camera coordinate frame, and the v-axis aligned with the negative z-axis of the camera coordinate frame.

B. EULER ANGLES

Euler angles are the classical means of representing rotations in 3-dimensional Euclidean space. The triplet of Euler angles (ϕ, θ, ψ) relates two orthogonal coordinate systems having a common origin. Conventionally, when the Euler angles (ϕ, θ, ψ) are used to describe the orientation of the aircraft body in relation to the inertia frame, these Euler angles are known as roll, pitch and yaw. (All information on Euler Angles is taken from Ref 3, Ref 4 and Ref 6.)

C. COORDINATE FRAMES RELATIONSHIPS

A coordinate transformation is a conversion from one coordinate system to another used to describe the same space. A rotation is a type of transformation from one system of coordinates to another system of coordinates such that distance between any

two points remains invariant under the transformation. A rigid body position in space can be represented by a [3 x 1] vector, and its orientation to its own current coordinate frame or a transformed coordinate frame can be uniquely described by a [3 x 3] rotation matrix at any instant in time. (All information on coordinate frame relationships is taken from Ref 2 and Ref 3.)

1. Rotation Matrices

The rotation matrices for a single 2 dimensional rotation about each individual axis are given below[1]. The angle of rotation is the Euler angle that corresponds to each individual axis.

$$R_Z(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$R_Y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (2)$$

$$R_X(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \quad (3)$$

The complete rotation or transformation of the coordinate system is the sequential combination of the 2 dimensional rotations about each axis. The sequence, or order of the rotation, is necessary to properly define the orientation of the body and to preserve the orthogonality orientation (Right hand system or Left hand system) of the transformed axes.

2. Coordinates Transformation

The coordinate transformation, or rotation from the inertia frame to the camera frame, can be obtained via sequential coordinate transformations from one frame to the other in the correct logical order, as shown below[2]:

[1] Ref 3, pp. 21-22, Equation (2.9), (2.10) and (2.11)

[2] Ref 3, p. 23, Equation in Figure 2.5, Transformation Sequence

$$ {}^{C}_{I}R = {}^{C}_{G}R\,{}^{G}_{B}R\,{}^{B}_{I}R \qquad (4) $$

where ${}^{B}_{I}R$ is the coordinate rotation from iniertia frame to body frame
${}^{G}_{B}R$ is the coordinate rotation from body frame to gimbal frame
${}^{C}_{G}R$ is the coordinate rotation from gimbal frame to camera frame

a. Inertial Frame to Body Frame Transformation

The coordinate transformation from the inertial frame to the body frame is simply the product of the three individual rotation matrices[3].

$$ {}^{B}_{I}R = R_x\left({}^{I}\phi_B\right)R_y\left({}^{I}\theta_B\right)R_z\left({}^{I}\psi_B\right) \qquad (5) $$

b. Body Frame to Gimbal Platform Frame Transformation

The coordinate transformation from the body frame to the gimbal platform frame only involves rotation through two angles because the gimbal platform is a two axis coordinate system[4]. The roll axis of the gimbal coordinate is fixed, and thus there is no roll rotation of the gimbal platform frame; i.e., the gimbal roll angle is zero.

$$ {}^{G}_{B}R = R_y\left({}^{B}\theta_G\right)R_z\left({}^{B}\psi_G\right) \qquad (6) $$

c. Gimbal Platform Frame to Camera Frame Transformation

The coordinate transformation between the gimbal platform frame and the camera frame allows for compensation of any misalignment angles that exist between the mounting of the camera platform with the gimbal platform[5]; in all likelihood, some or all of the rotation angles will be zero because the axes will be directly aligned.

$$ {}^{C}_{G}R = R_x\left({}^{G}\phi_C\right)R_y\left({}^{G}\theta_C\right)R_z\left({}^{G}\psi_C\right) \qquad (7) $$

[3] Ref 3, p 35, Equation (2.35)

[4] Ref 3, p 37, Equation (2.36)

[5] Ref 2, p 8, Equation (7)

d. Camera Frame to Image Plane Frame Transformation

The coordinate transformation between the camera frame and the image plane frame is not a rotational but a positional transformation. The position of an object in the image plane frame from a position in the camera frame is given as below[6]:

$$\begin{bmatrix} P_u \\ P_v \end{bmatrix} = \frac{f}{c_x} \begin{bmatrix} c_y \\ -c_z \end{bmatrix} \tag{8}$$

It is of interest to note that, unlike the previous rotational transformations in the earlier coordinate transformation schemes, which are reversible, the Camera to Image Plane transformation is irreversible; this is due to the transformation of a three dimensional coordinate system to a two dimensional coordinate system.

3. Angular Velocities Transformations

a. Body Frame Angular Velocities with Respect to Inertia Frame

The angular velocity vector ω, in the body fixed coordinates system of the SUAV, has components [p, q, r] in the x, y and z direction, respectively. The resulting relationships with the Euler angle rates $\left[\dot{\phi}_B, \dot{\theta}_B, \dot{\psi}_B\right]$ are stated as follows[7]:

$$\begin{aligned} {}^{B}\omega_{BI} &= R_x\left({}^{I}\phi_B\right) R_y\left({}^{I}\theta_B\right) R_z\left({}^{I}\psi_B\right) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi}_B \end{bmatrix} + R_x\left({}^{I}\phi_B\right) R_y\left({}^{I}\theta_B\right) \begin{bmatrix} 0 \\ \dot{\theta}_B \\ 0 \end{bmatrix} + R_x\left({}^{I}\phi_B\right) \begin{bmatrix} \dot{\phi}_B \\ 0 \\ 0 \end{bmatrix} \\ {}^{B}\omega_{BI} &= \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} p_B \\ q_B \\ r_B \end{bmatrix} = \begin{bmatrix} \dot{\psi}_x + \dot{\theta}_x + \dot{\phi}_x \\ \dot{\psi}_y + \dot{\theta}_y + \dot{\phi}_y \\ \dot{\psi}_z + \dot{\theta}_z + \dot{\phi}_z \end{bmatrix} = \begin{bmatrix} \dot{\phi}_B - \dot{\psi}_B \sin {}^{I}\theta_B \\ \dot{\psi}_B \cos {}^{I}\theta_B \sin {}^{I}\phi_B + \dot{\theta}_B \cos {}^{I}\phi_B \\ \dot{\psi}_B \cos {}^{I}\theta_B \sin {}^{I}\phi_B - \dot{\theta}_B \cos {}^{I}\phi_B \end{bmatrix} \end{aligned} \tag{9}$$

Expressing $\left[\dot{\phi}_B, \dot{\theta}_B, \dot{\psi}_B\right]$ in terms of $[p_B, q_B, r_B]$ below, we observe the singular problem in equation (10) when ${}^{I}\theta_B = \pm 90^o$.

---

[6] Ref 2, p 8, Equation (8)

[7] Ref 3, pp 26, Equation (2.20) and (2.21)

$$\begin{bmatrix} \dot{\psi}_B \\ \dot{\theta}_B \\ \dot{\phi}_B \end{bmatrix} = \begin{bmatrix} \frac{1}{\cos\left({}^I\theta_B\right)}\left[q_B \sin\left({}^I\phi_B\right) + r_B \cos\left({}^I\phi_B\right)\right] \\ q_B \cos\left({}^I\phi_B\right) - r_B \sin\left({}^I\phi_B\right) \\ p_B + \tan\left({}^I\theta_B\right)\left[q_B \sin\left({}^I\phi_B\right) + r_B \cos\left({}^I\phi_B\right)\right] \end{bmatrix} \tag{10}$$

b. Gimbal Frame Angular Velocities with Respect to Inertia Frame

The angular velocities of the gimbal frame with respect to the inertia frame are shown below. Equation (10) relates the angular velocity expressed in the gimbal platform frame while equation (11) describes the transformation of the same angular rate in the inertia frame[8].

$${}^G\omega_{GI} \triangleq \begin{bmatrix} p_G \\ q_G \\ r_G \end{bmatrix} = R_y\left({}^I\theta_G\right) R_z\left({}^I\psi_G\right) \begin{bmatrix} p_B \\ q_B \\ r_B \end{bmatrix} + R_y\left({}^I\theta_G\right) R_z\left({}^I\psi_G\right) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi}_G \end{bmatrix} + R_y\left({}^I\theta_G\right) \begin{bmatrix} 0 \\ \dot{\theta}_G \\ 0 \end{bmatrix}$$

$${}^G\omega_{GI} = {}^G_B R \begin{bmatrix} p_B \\ q_B \\ r_B \end{bmatrix} + {}^G_B R \begin{bmatrix} 0 \\ 0 \\ \dot{\psi}_G \end{bmatrix} + R_y\left({}^I\theta_G\right) \begin{bmatrix} 0 \\ \dot{\theta}_G \\ 0 \end{bmatrix} \tag{11}$$

$${}^I\omega_{GI} = {}^I_G R\, {}^G\omega_{GI} = {}^I_G R \begin{bmatrix} p_G \\ q_G \\ r_G \end{bmatrix} \tag{12}$$

c. Gimbal Frame Angular Velocities with Respect to Inertia Frame

If the camera frame is perfectly aligned with the gimbal platform frame, then there will be no rotation between the two frames. Therefore, the angular rates between the two frames will be the same. The relationships are expressed as below[9]:

[8] Ref 2, p 9, Equation (9)

[9] Ref 2, p 10, Equation (10) and (11)

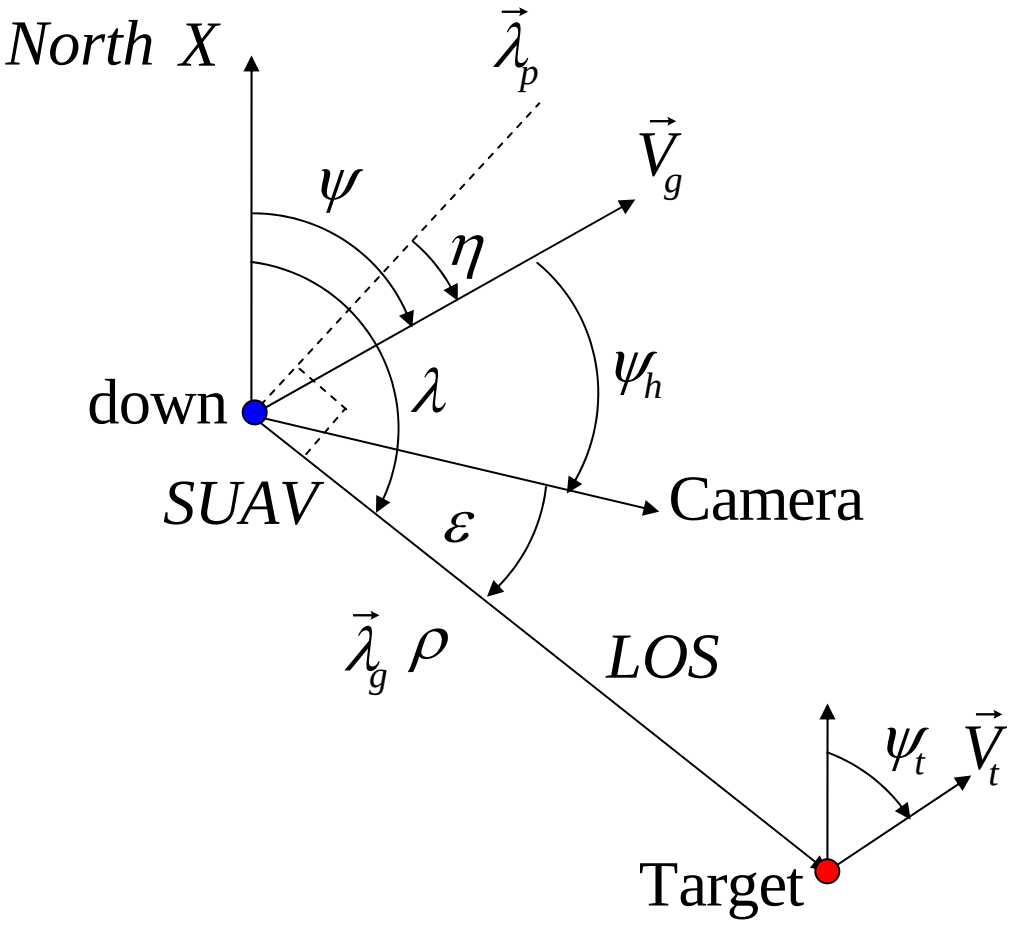
$$^{C}\omega_{CI} = {}^{C}_{G}R\,{}^{G}\omega_{GI} \qquad ; {}^{C}_{G}R = I$$

$$^{C}\omega_{CI} = {}^{G}\omega_{GI} = \begin{bmatrix} p_G \\ q_G \\ r_G \end{bmatrix} \tag{13}$$

$$^{I}\omega_{CI} = {}^{I}\omega_{GI} \tag{14}$$

D. KINEMATICS EQUATIONS OF THE SUAV-TARGET MOTION

The simplified two-dimensional (2D) kinematics model is presented in Figure 4; it is assumed that an autopilot can easily hold altitude, therefore converting the task to 2D. The figure depicts the relationships between the kinematics angles used to characterize relative motion of SUAV-Target.



η	= navigation error	ψ	= UAV heading in inertia frame
ε	= camera LOS Pan error	ψ_t	= Target heading in inertia frame
λ	=LOS angle in inertia frame	ψ_h	= Camera angle in UAV body frame
$\vec{\lambda}_g$	= LOS vector	$\vec{V}_g$	= SUAV ground speed in inertia frame
$\vec{\lambda}_p$	= Normal to LOS vector	$\vec{V}_t$	= Target ground speed in inertia frame

Figure 4: Kinematics of the SUAV—Target Motion

From the kinematics above, we derive the SUAV-target kinematics' equations, as shown below:

$$\begin{aligned}
\dot{\eta} &= -\frac{V_g \cos\eta - V_t \cos\left(\psi_t - \left(\psi - \eta\right)\right)}{\rho} + \dot{\psi} \\
\dot{\varepsilon} &= \frac{V_g \cos\eta - V_t \cos\left(\psi_t - \left(\psi - \eta\right)\right)}{\rho} - \dot{\psi} - \dot{\psi}_h \\
\dot{\rho} &= -V_g \sin\eta + V_t \sin\left(\psi_t - \left(\psi - \eta\right)\right)
\end{aligned} \qquad (15)$$

The navigation angle error η is the angle subscribed between the SUAV's ground velocity vector and the perpendicular to LOS vector. The navigation angle error rate approaches zero when the SUAV establishes a circular orbit about a target at the desired range.

The camera LOS angle error is the angle between the camera middle line and the LOS vector. The camera LOS angle rate approaches zero when the camera LOS angle is pointing either 90^o left or right of the SUAV's ground velocity vector (when the SUAV is in a circular orbit around a target at the desired range).

E. CONTROL LAW DESIGN

A suitable form of the control laws for implementation on the SUAV AP Controller and Gimbal Platform controller is shown as follows:

$$\begin{aligned}
\dot{\psi} &= \frac{V_g}{\rho_d}\cos\eta - k_1\eta \\
\dot{\psi}_h &= k_1\eta + k_2\varepsilon
\end{aligned} \qquad (16)$$

η = navigation angle error
ε = camera LOS angle error
V_g = SUAV ground speed
$k_{1,} k_2$ = gain constant
$\dot{\psi}$ = UAV turn rate in inertia frame
$\dot{\psi}_{\mathrm{h}}$ = Camera turn rate in UAV body frame
ρ_d = desired range

The chosen form of control law will allow for dynamic adjustment of the required turn rate of the SUAV ($\dot{\psi}$); this acts in accordance to the magnitude of the ground velocity vector of the SUAV to converge of its desired range. If the SUAV begins by

tracking a stationary target inside the desired range, it will spiral outwards to the desired range; if the SUAV begins by tracking the target outside the desired range, it will spiral inwards to the desired range. When the SUAV is established at its desired range in a circular orbit, the turn rate of the SUAV will approach the required turning "bias", which will keep the SUAV in circular orbit around the target at the desired range. The gimbal turn rate of the gimbaled camera will also approach zero when the SUAV has established the +/- 90^o camera LOS angle in the circular orbit at the desired range.

The nonlinear control law in Equation (16) includes an *interesting feature*. As it is shown below in (17), the control law is able to drive the range to target ρ to the desired value ρ_d. This is done for the *unknown* ρ. The control intuition suggests that this can be achieved by driving the SUAV's yaw rate to the desired value $\frac{V_g}{\rho_d}$.

By substituting control law (16) into the kinematics equation (15) and performing some algebraic manipulations, we discover the following feedback system:

$$\begin{aligned}
&\dot{\eta} = -V_g \rho_e \cos\eta - k_1 \eta + V_t \cos(\eta - \psi)\bar{\rho} \\
&\dot{\varepsilon} = V_g \rho_e \cos\eta - k_2 \varepsilon - V_t \cos(\eta - \psi)\bar{\rho} \\
&\dot{\rho}_e = \bar{\rho}^2 V_g \sin\eta - V_t \sin(\eta - \psi)\bar{\rho}^2 \\
&\textit{where} \quad \rho_e = \frac{1}{\rho} - \frac{1}{\rho_d} \quad \Rightarrow -\frac{1}{\rho^2}\dot{\rho} = \dot{\rho}_e; \quad \frac{1}{\rho} = \bar{\rho}
\end{aligned} \tag{17}$$

It becomes apparent from equation (17) that, by driving the navigation angle error (η) and camera LOS angle error (ε) to zero, the range error ρ_e is indirectly driven to zero. Therefore, although range to target is not measured directly, the proposed control law allows us to control it.

F. EIGENVALUE STABILITY ANALYSIS OF THE FEEDBACK SYSTEM

For purposes of stability analysis, it is convenient to rescale η by introducing a new state variable $\hat{\eta} = \frac{\eta}{\rho_d}$, substituting in equation (17), and rearranging:

$$\dot{x} := \begin{bmatrix} \dot{\hat{\eta}} \\ \dot{\rho}_e \\ \dot{\varepsilon} \end{bmatrix} = \begin{bmatrix} -V_g \rho_e \cos\hat{\eta}\rho_d - k_1 \hat{\eta}\rho_d + V_t \cos(\hat{\eta}\rho_d - \psi)\bar{\rho} \\ \bar{\rho}^2 V_g \sin\hat{\eta}\rho_d - V_t \cos(\hat{\eta}\rho_d - \psi)\bar{\rho}^2 \\ V_g \rho_e \cos\hat{\eta}\rho_d - k_2 \varepsilon - V_t \cos(\hat{\eta}\rho_d - \psi)\bar{\rho} \end{bmatrix} \tag{18}$$

where $x = \begin{bmatrix} \hat{\eta} & \rho_e & \varepsilon \end{bmatrix}^T$

Using Eigenvalue analysis, the stability of the system (18) is next addressed. If we first suppose that the target is stationary, then $V_t = 0$. Consequently,

$$\dot{x} := \begin{bmatrix} \dot{\hat{\eta}} \\ \dot{\rho}_e \\ \dot{\varepsilon} \end{bmatrix} = \begin{bmatrix} -V_g \rho_e \cos\hat{\eta}\rho_d - k_1 \hat{\eta}\rho_d \\ \bar{\rho}^2 V_g \sin\hat{\eta}\rho_d \\ V_g \rho_e \cos\hat{\eta}\rho_d - k_2 \varepsilon \end{bmatrix}, \tag{19}$$

and the origin $x = (0\ 0\ 0)$ is clearly the equilibrium of (19). Linearization of (19) around the origin yields an LTI system

$$\dot{\xi} = \begin{bmatrix} -k_1 & -V_g / \rho_d & 0 \\ V_g / \rho_d & 0 & 0 \\ 0 & V_g & -k_2 \end{bmatrix} \xi \tag{20}$$

We may then also assume that the SUAV velocity V_g is constant, and $V_g \in \left[V_{g_{\min}}, V_{g_{\max}}\right]$, $V_{g_{\max}} \geq V_{g_{\min}} > 0$. Then the eigenvalues of the state matrix in (20) have negative real parts for any $k_1 > 0,\ k_2 > 0$. Therefore, the nonlinear system (19) is locally asymptotically stable for any $k_1 > 0,\ k_2 > 0$.

On the other hand, if the target is moving and $V_t \neq 0$, the equilibrium of (18) is at the relative heading $\psi = \frac{\pi}{2}$. This in fact corresponds to the circular motion of the SUAV around the target. In this case, linearization of (18) around the equilibrium results in an LTI system

$$\dot{\xi}_1 = \begin{bmatrix} -k_1 + V_t/\rho_d & -V_g/\rho_d & 0 \\ (V_g - V_t)/\rho_d & 0 & 0 \\ -V_t & V_g & -k_2 \end{bmatrix} \xi_1 \qquad (21)$$

The target velocity V_t is assumed constant, and $V_t \in \left[V_{t_{\min}}, V_{t_{\max}}\right]$, $V_{t_{\max}} \geq V_{t_{\min}} > 0$. The eigenvalues of the LTI system (21) will have negative real parts if $V_g > V_t$, $k_1 > V_t/\rho_d$ and $k_2 > 0$. As a result, if these conditions hold, the nonlinear system (17) is locally asymptotically stable. The Eigenvalue analysis plot for V_t = 20m/s is shown as follows (Fig.5):

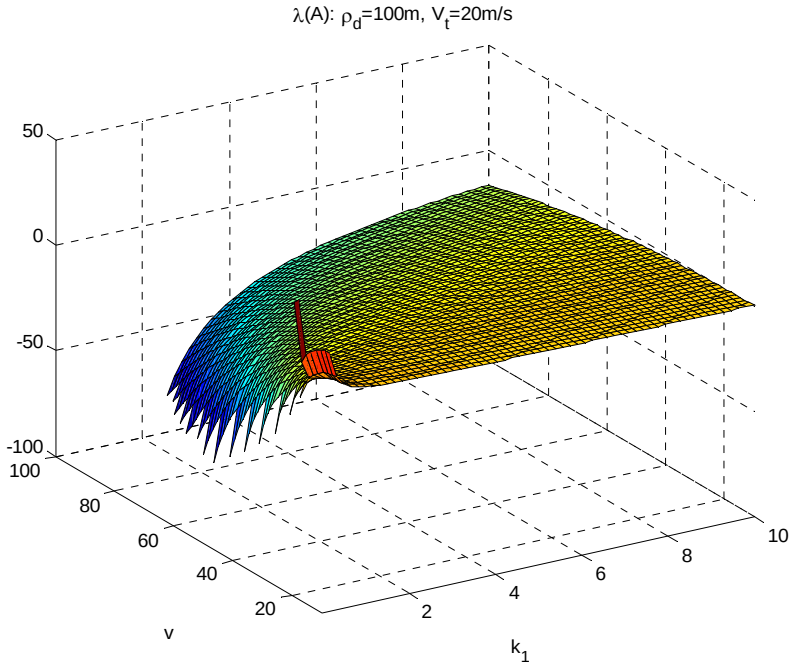


Figure 5: Eigenvalue Analysis Plot

The plot demonstrates that the eigenvalues of the state matrix in (21) are always negative, except the area where V_g is less than V_t. The Eigenvalues analysis suggests that the feedback system is locally asymptotically stable.

The plots of the state trajectories of system (17), in response to a number of initial conditions, are included as follows (Fig. 6):

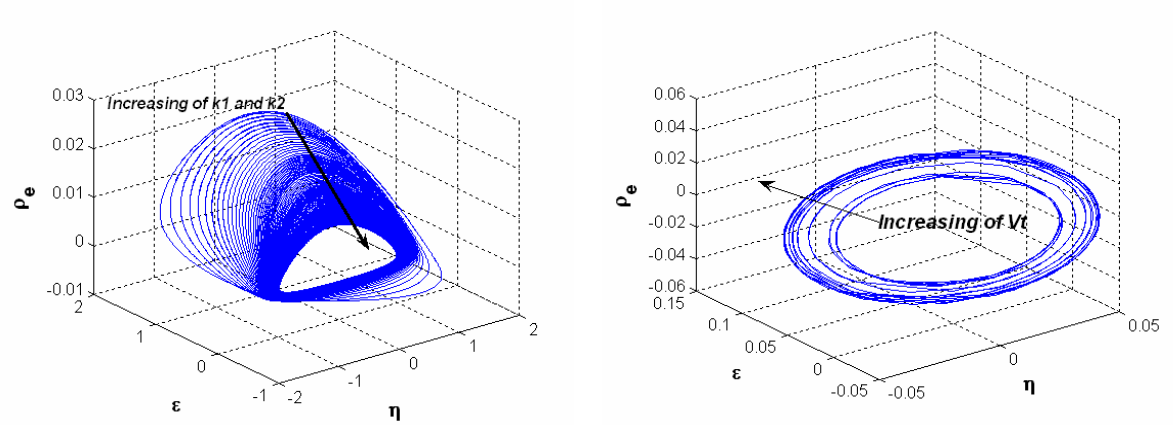


Figure 6: Steady State Trajectories

The plot on the left illustrates the impact of increasing the gains k_1, k_2.for a fixed V_t on the trajectories of feedback system (17). Conversely, the plot on the right demonstrates the influence of increasing V_t for fixed k_1 and k_2. The figures demonstrate that the trajectories of feedback system (17) converge to a ball, the size of which is proportional to V_t and inversely proportional to k_1 and k_2. (Ref 7)

The Eigenvalue stability analysis for the feedback system is coded as MATLAB Symbolic Toolbox script file; the details can be found in appendix A.

THIS PAGE INTENTIONALLY LEFT BLANK

III. CONTROL LAW IMPLEMENTATION

A. MODIFICATIONS OF SIMULINK MODEL FOR IMPLEMENTATION OF NEW CONTROL LAW

The initial control law (Ref 1) of the VBTT system has the following form:

$$\begin{aligned} \dot{\psi} &= sign\left(Bias - k_1\eta\right) \\ \dot{\psi}_h &= k_2\varepsilon \end{aligned} \tag{22}$$

Two observations are worthy of mention here. First, the turning bias in the initial control law (See equation (22)) is a fixed value, and thus does not vary with the SUAV ground speed or the desired range. Second, the camera LOS turn rate is independent of the SUAV turn rate.

The fixed bias value applied to turn the SUAV in equation (22) can be any arbitrary value. As a consequence, it is slow in converging to any other desired range commanded, except to the desired range that corresponds to the arbitrary turning bias value. In addition, because the camera LOS turns independently of the SUAV turn rate, the resulting control of the camera LOS is poor, and necessitates a large k_2 value to keep the camera LOS closely aligned to the SUAV-target LOS.

Using the newly designed control law in equation (16), equation (22) adjusts to the following form:

$$\begin{aligned} \dot{\psi} &= \frac{\vec{V}_g}{\rho_d}\cos\eta - k_1\eta \\ \dot{\psi}_h &= k_1\eta + k_2\varepsilon \end{aligned} \tag{23}$$

The turning bias in equation (22) is replaced with the dynamically adjusted quantity in equation (23), which varies with the UAV ground speed and the desired range. The control law in equation (23) improves upon equation (22) in that the dynamically adjusted quantity replicates the “old fixed bias” term in equation (22) and

automatically adjusts turn rate command to the target motion. In addition, the camera LOS turn rate is coupled and compensated with the SUAV turn rate, which makes the camera control more efficient.

The control system architecture that implements control law (23) is presented in Figure 7. It consists of an autopilot and a gimbal driven by the control inputs $\dot{\psi}$ and $\dot{\psi}_h$. Onboard cameras provide real-time imagery to the image tracking software. In turn, the software computes the tracking error ε, while onboard GPS and inertial systems provide solution for the navigation error η. (Ref 7)

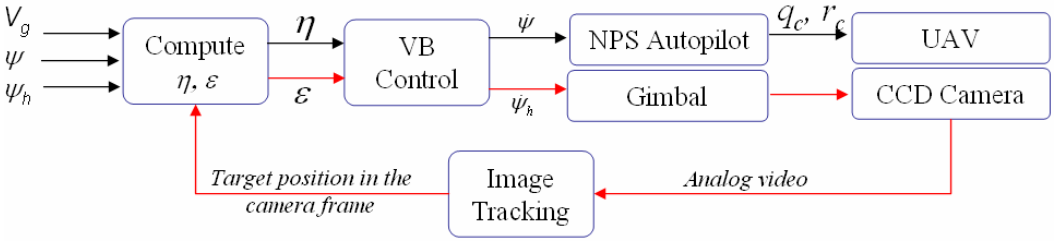


Figure 7: Control System Architecture

The VB Control block pictured in Figure 7 is the actual implementation of the control law (23).

The control law for the Gimbaled camera yaw command $\dot{\psi}_h$ in equation (20) is implemented in SIMULINK between the "Gimbaled Camera model" and "CurGuid Controller" blocks (shown in Figure 8).

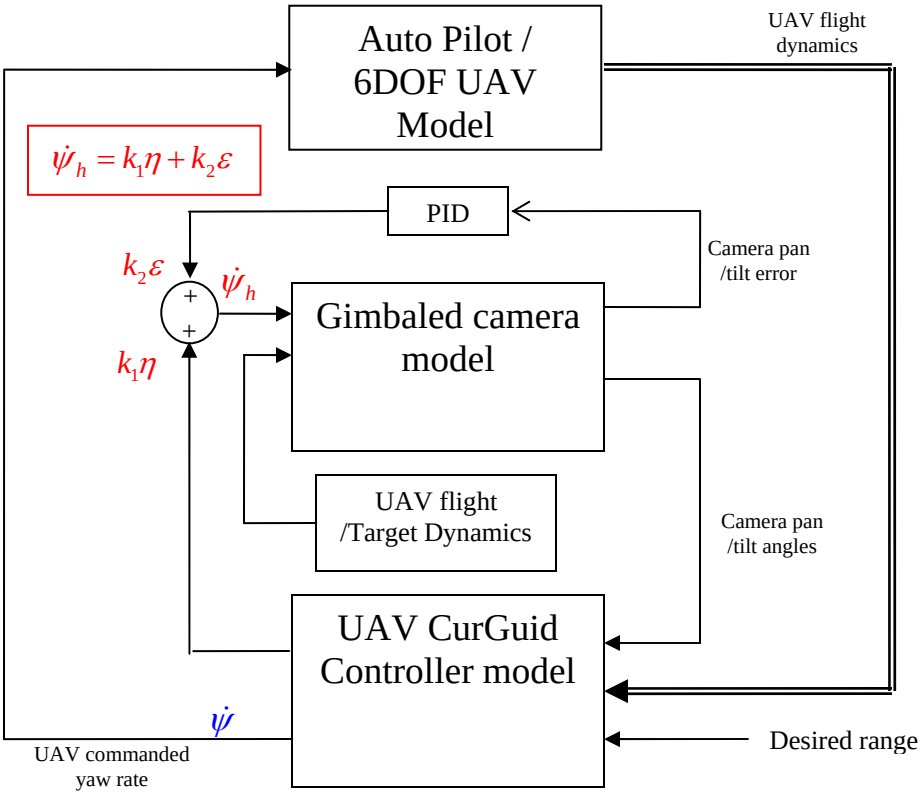


Figure 8: VBTT SIMULINK Model Schematic

The control law for the SUAV yaw rate command $\dot{\psi}$ in equation (20) is implemented in the "CurGuid Controller" block (shown in Figure 9).

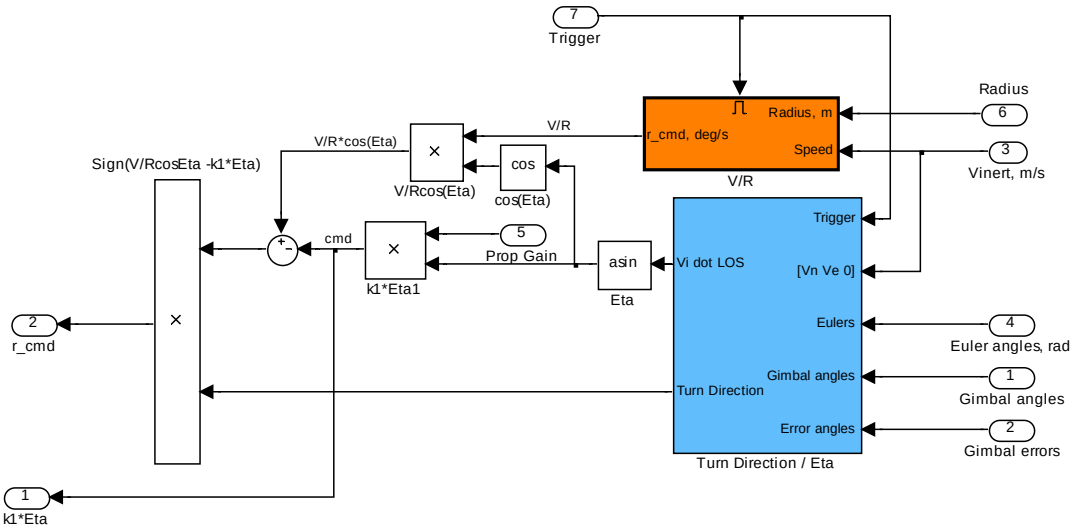


Figure 9: SIMULINK CurGuid Controller Block

B. SIMULATION RESULTS

The performance of the newly developed control law is tested against both a stationary target and a moving target under different target motion scenarios. From these, the sensitivity analysis results of the new control law are derived for feedback control parameters k_1 and the relative velocity ratio of the target and SUAV, $\frac{V_t}{V_g}$.

Since the parameter k_2 affects only the gimbal control law (see second equation in (23)), it is not examined for the cross coupled sensitivity between the SUAV yaw command and the camera LOS yaw command. The camera controller is independent of SUAV guidance. By appropriate choice of k_2, the camera controller loop can be chosen to be much faster than the SUAV guidance loop.

Two measures of performance (MOP-1 and MOP-2) are devised to assess the quality of the newly developed control law (23). The physical meanings and technical definitions of the MOPs are as follows:

MOP-1 is devised to measure the convergence speed to the desired range; this speed is affected by range and convergence time. Convergence time is defined as the time required for the SUAV to converge to its first zero crossing of ρ_e. The radial

distance that is covered by the SUAV during the convergence period is defined as the range capture. Therefore, MOP-1 is a ratio of the range capture over the convergence time. A higher MOP-1 value is desirable, as it represents a faster range capture capability.

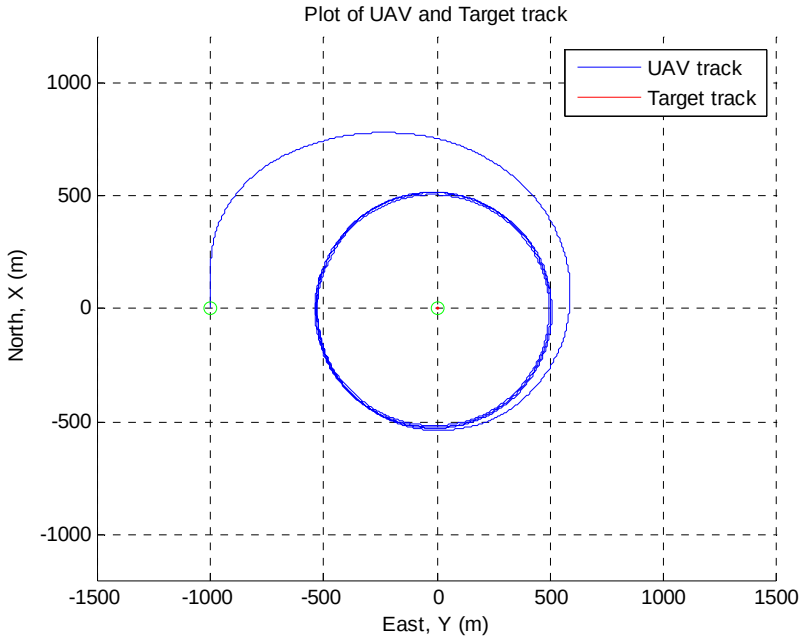
MOP-2 is devised to measure the range holding capability of the SAUV after it has captured the desired range. The maximum range deviation from the desired range after its first closest approach is utilized to measure the SUAV range holding capability at the desired range. MOP-2 is defined as the ratio of maximum deviation from the desired range to the commanded range, expressed in percent. A lower MOP-2 value is desirable, as it represents a better range holding capability of the SUAV.

1. Sensitivity Analysis of the Control Law to Variations of K_1

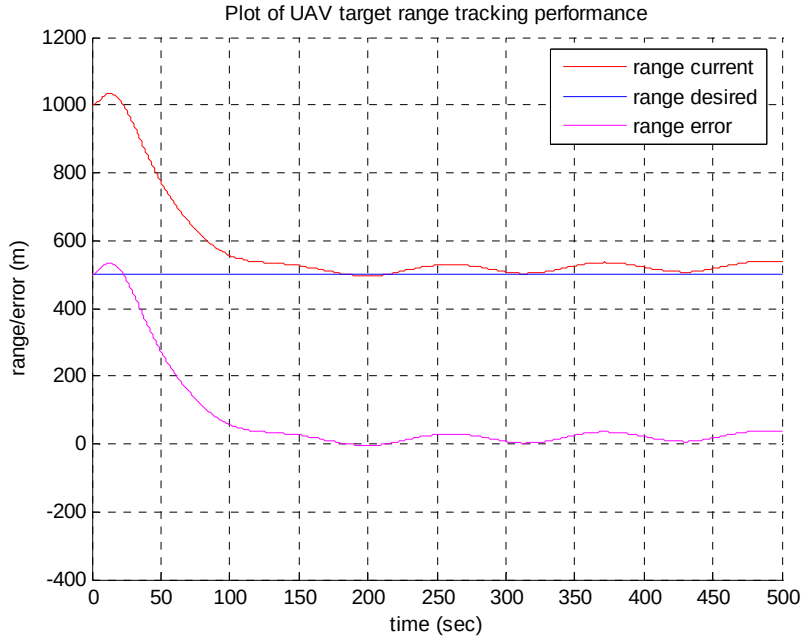
The sensitivity analysis of the control law to variations of parameter k_1 is examined in a scenario in which the desired range is 500m. The initial conditions are: (1) SUAV velocity = 28m/s, (2) Target velocity = 0 m/s, (3) Initial position of SUAV is at [0, -1000, 300], (4) Initial position of target is at [0, 0, 0] (The initial horizontal ground range from UAV to target is 1000m), and (5) $k_2 = 0.25$.

Case 1: K_1= 0.1

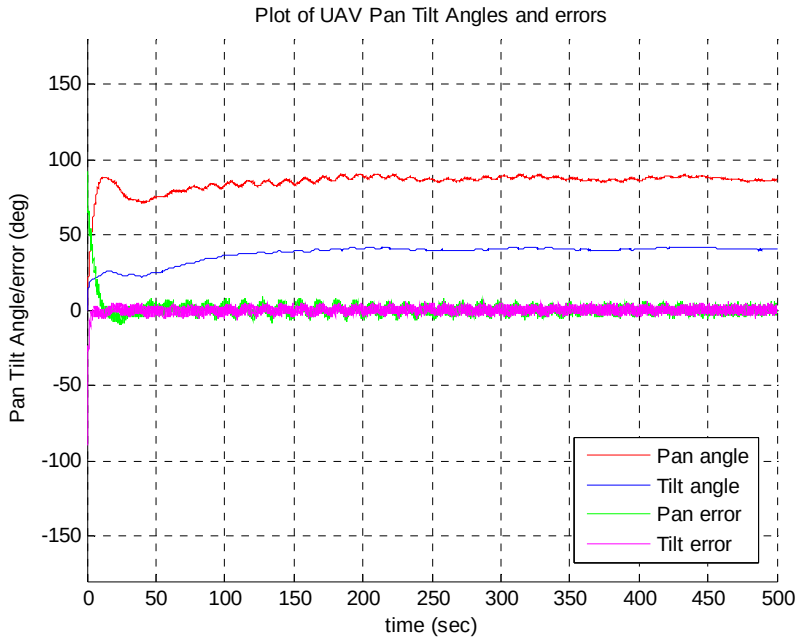
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 800m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

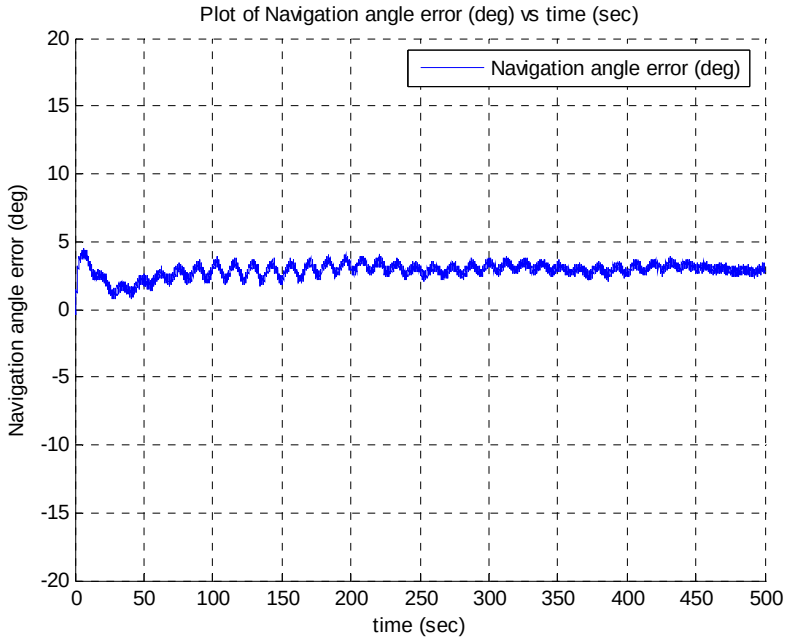
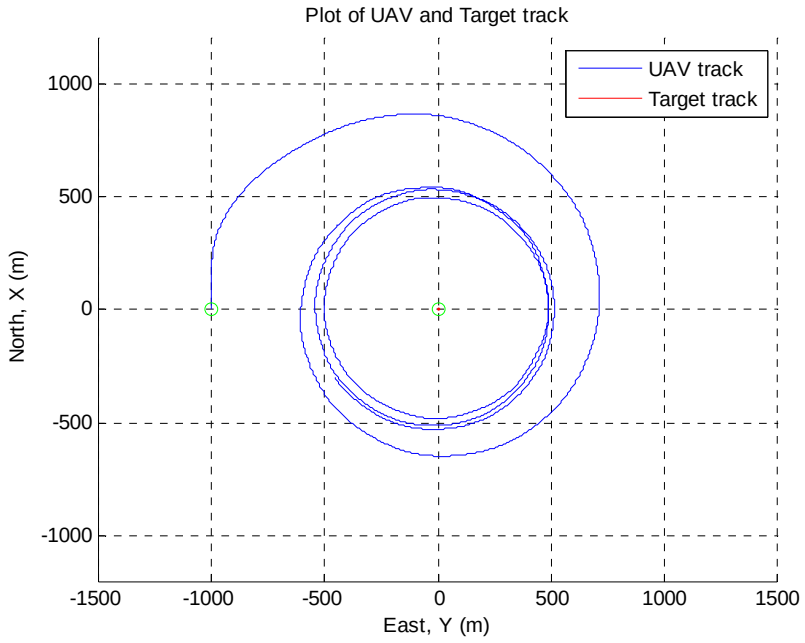


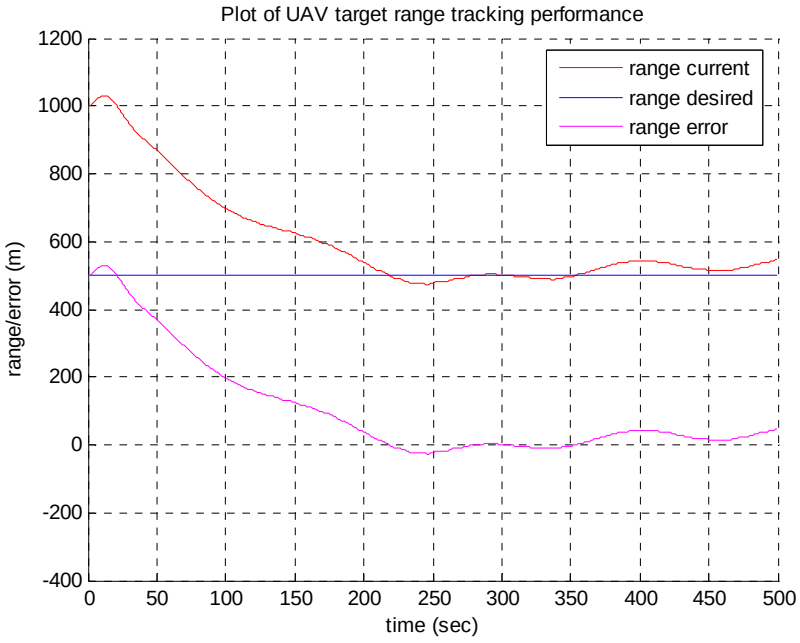
Figure 10: Sensitivity Analysis for K_1= 0.1 at Desired Range 500m

Case 2: K_1= 0.2

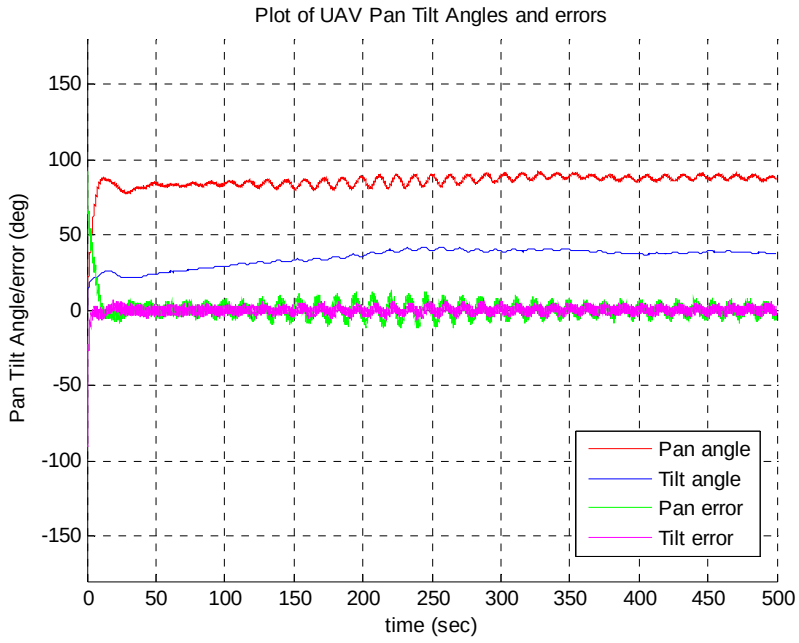
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 500m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

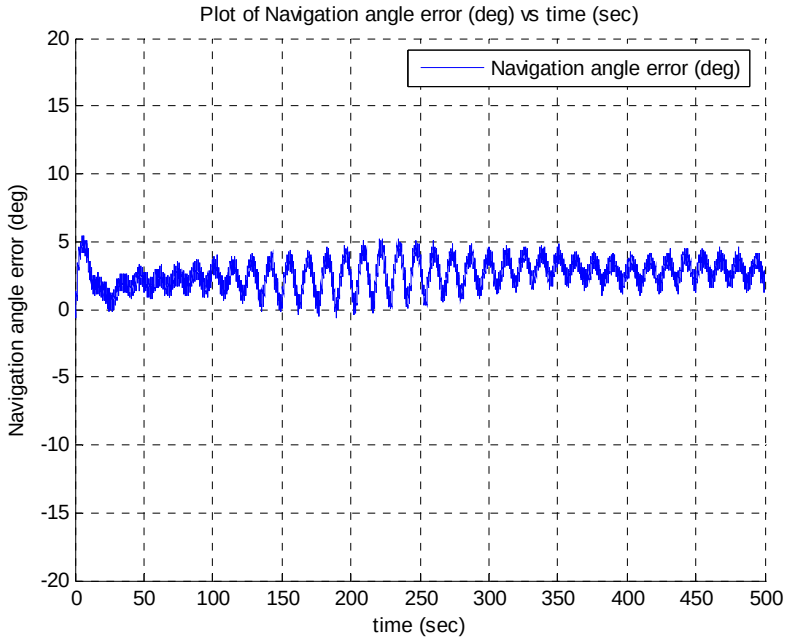
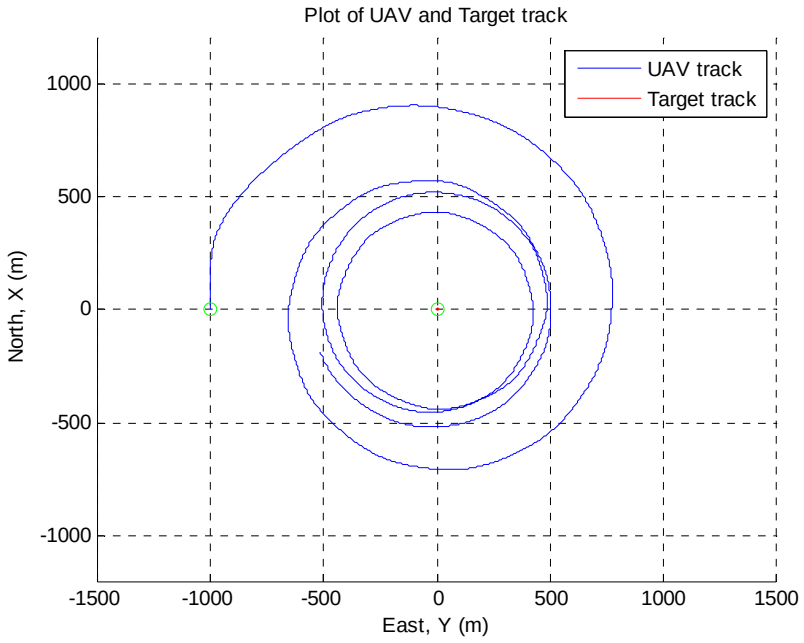


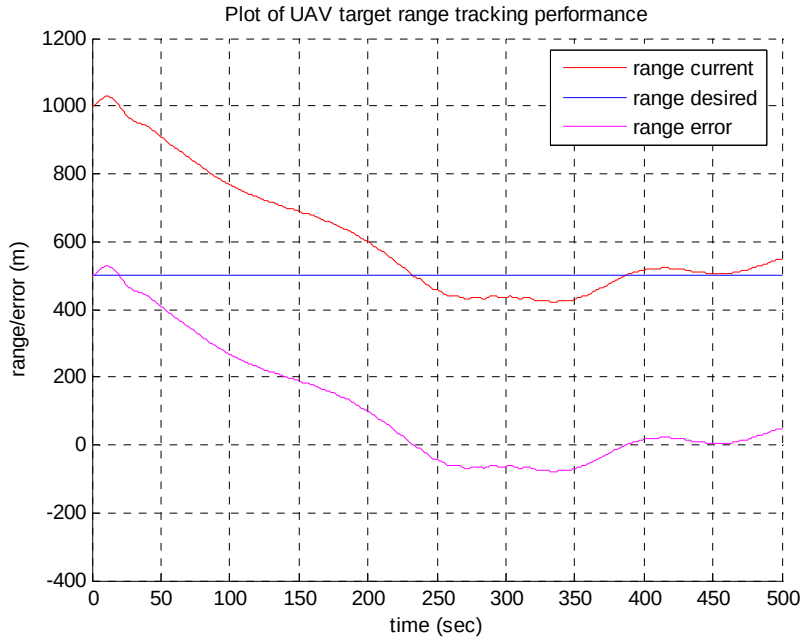
Figure 11: Sensitivity Analysis for K_1= 0.2 at Desired Range 500m

Case 3: K_1= 0.3

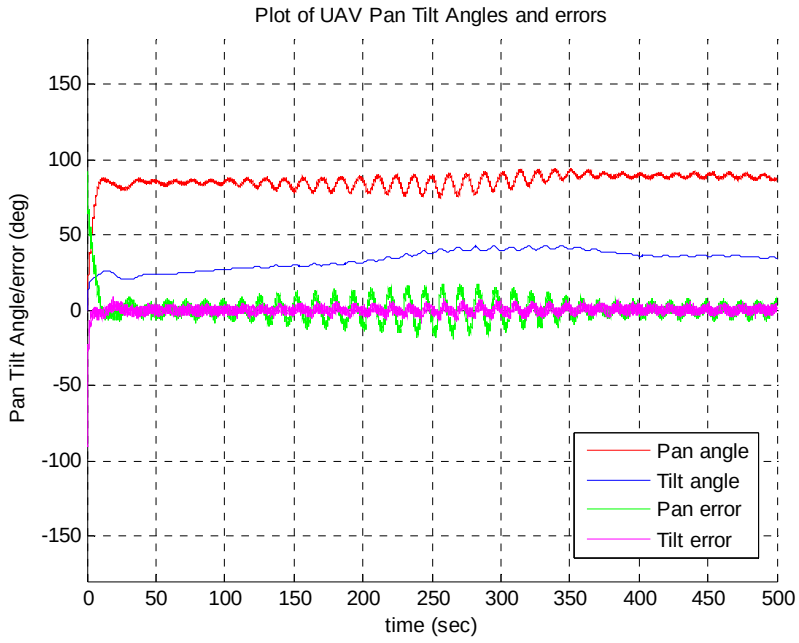
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 500m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

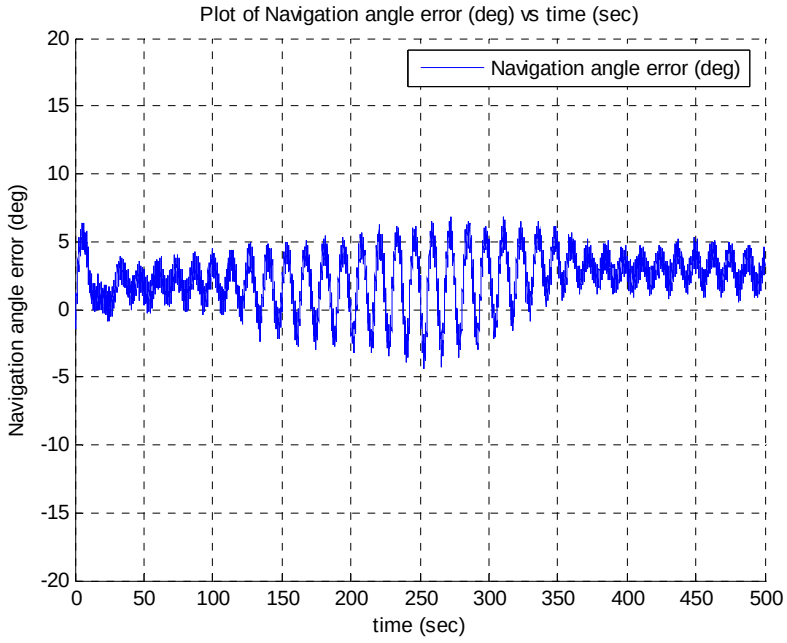
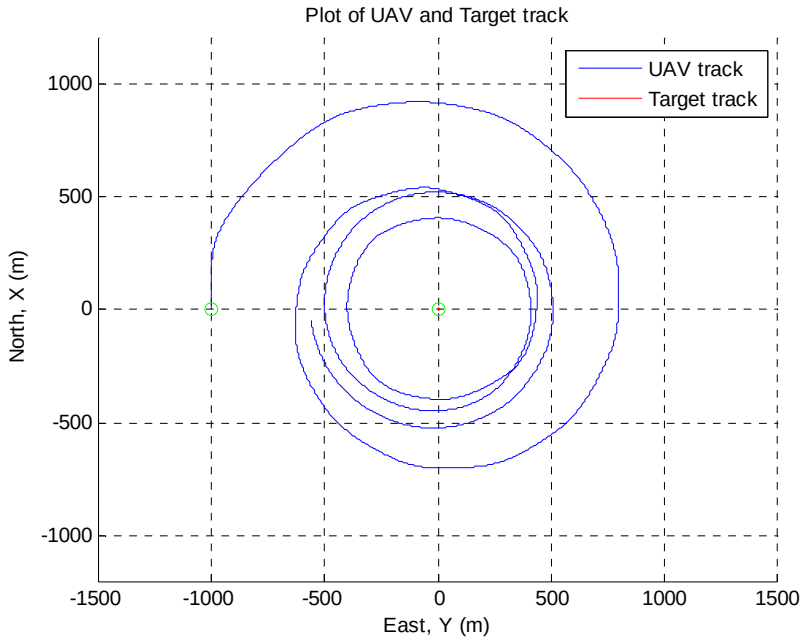


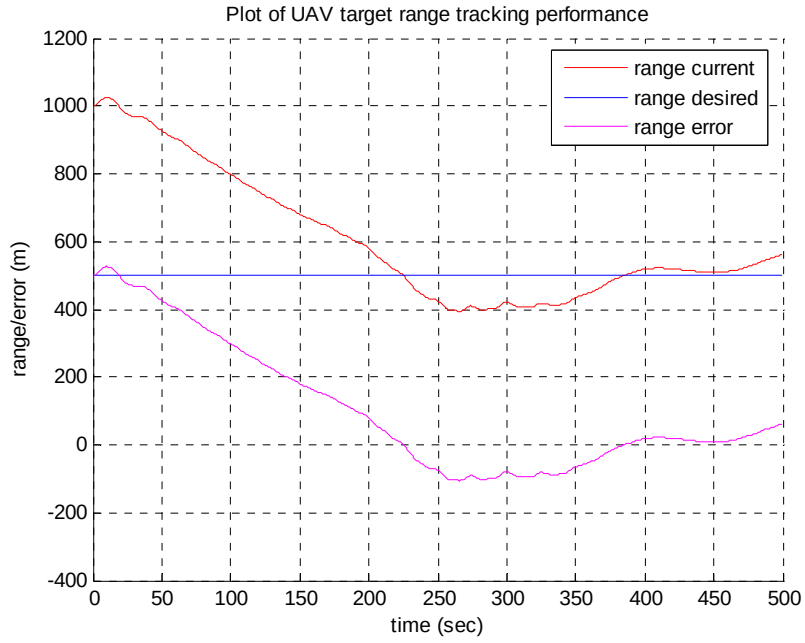
Figure 12: Sensitivity Analysis for K_1= 0.3 at Desired Range 500m

Case 4: K_1= 0.4

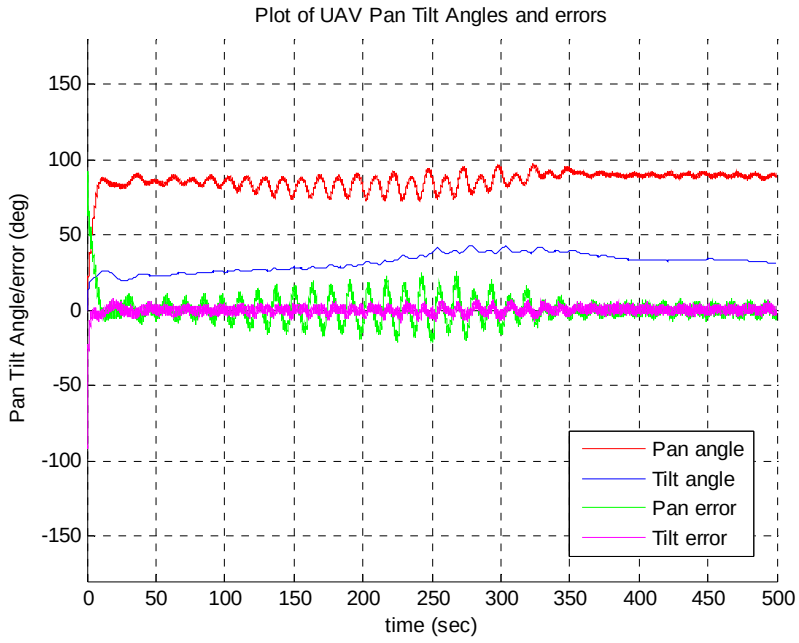
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 500m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

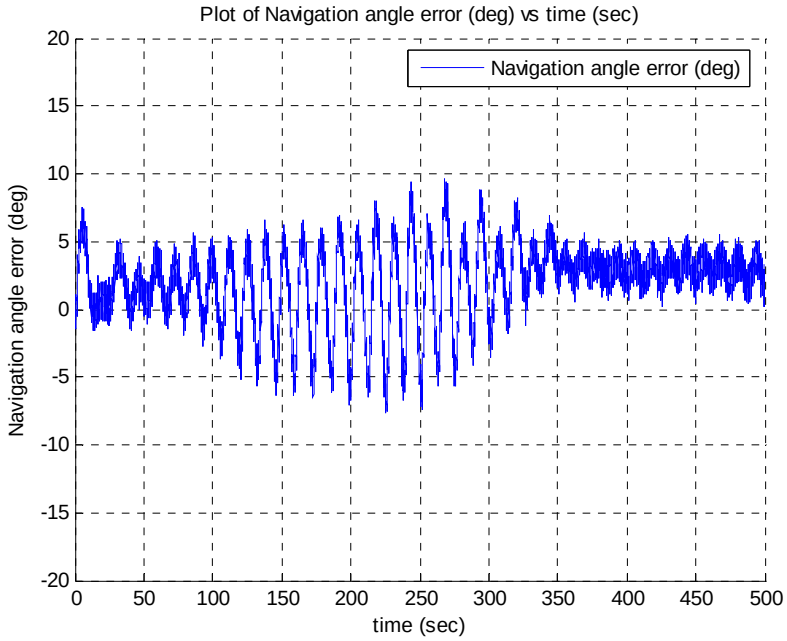


Figure 13: Sensitivity Analysis for K_1= 0.4 at Desired Range 500m

The contributors of MOP-1(k_1) for the four test cases are tabulated below:

K_1	Range Captured (m)	Time to first closest approach (sec)	MOP-1(k_1), convergence speed (m/sec)
0.1	500	180	2.778
0.2	500	220	2.273
0.3	500	230	2.174
0.4	500	225	2.222

Table 1: MOP-1(k_1)

The MOP-1(k_1) of the SUAV is plotted as follows:

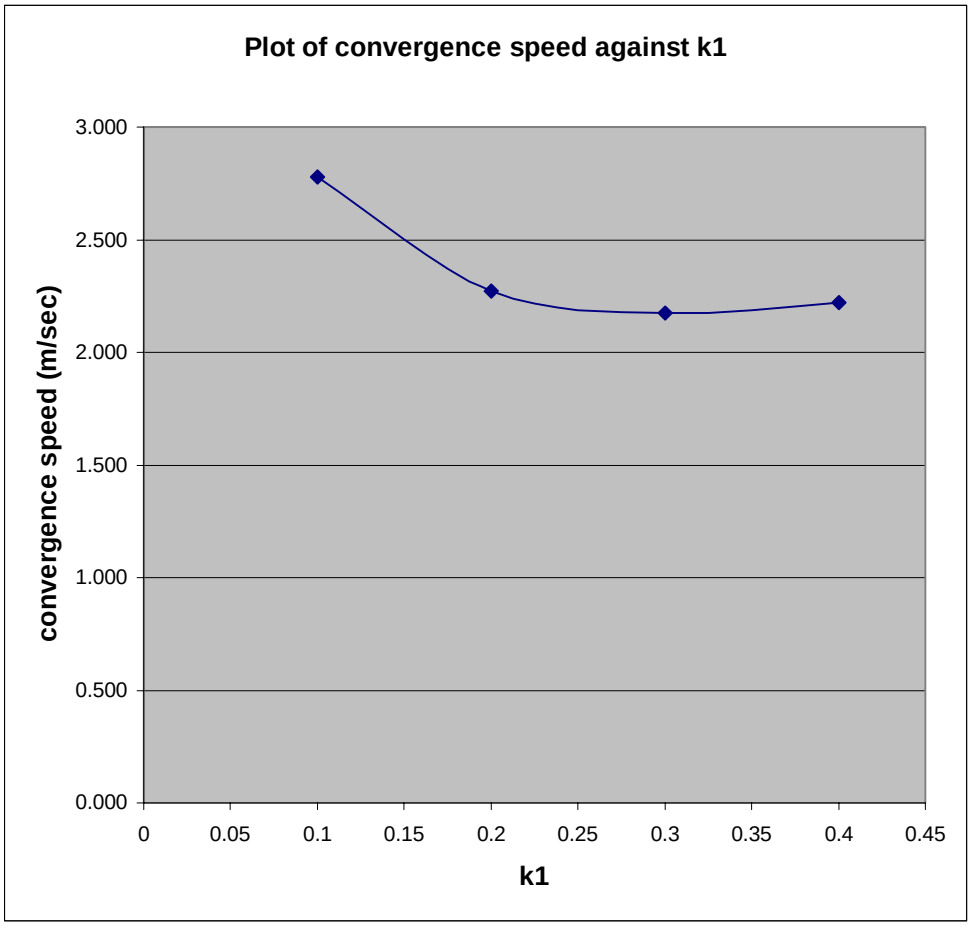


Figure 14: MOP-1(k_1)

The contributors of MOP-2(k_1) for the four test cases are tabulated below:

K_1	Maximum range deviation error (m)	MOP-2(k_1), Percentage deviation from desired range (%)
0.1	40	8.0
0.2	35	7.0
0.3	50	10.0
0.4	100	20.0

Table 2: MOP-2(k_1)

The MOP-2(k_1) of the SUAV is plotted as follows:

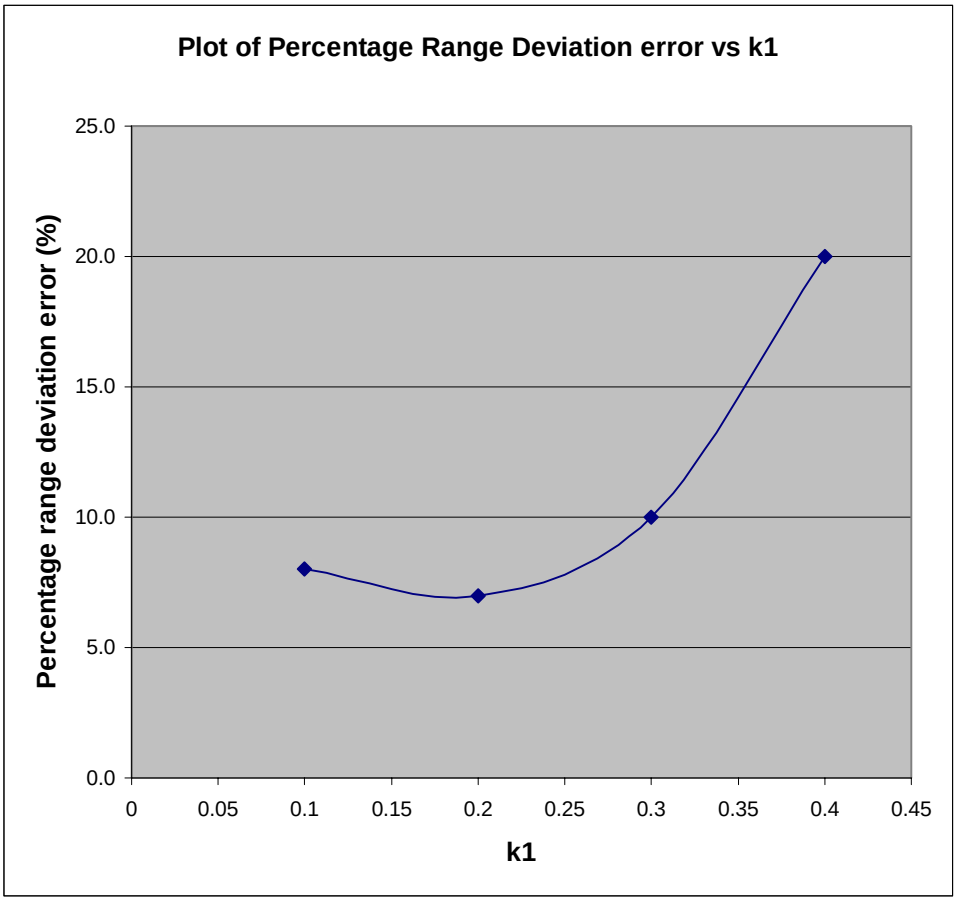


Figure 15: MOP-2(k_1)

The following observations can be made from the MOP plots and test plots above:

(a) From MOP-1(k_1) analysis, it can be observed that the convergence speed of the SUAV decreases and levels off to a plateau with increasing k_1. The highest value is 2.778 (m/sec) when k_1 is 0.1, and the lowest value is 2.174 (m/sec) when k_1 is 0.3.

(b) From MOP-2(k_1) analysis, it is observable that the percentage range deviation error of the SUAV is at its minimum at 7.0% when k_1 is 0.2; the maximum range deviation error is 20.0% when k_1 is 0.4. When k_1 is 0.2, the range holding performance of the SUAV decreases rapidly with increasing k_1 values after the minimum value.

(c) From the test plots (Fig. 10 – 14) above, it can be observed that an increase in k_1 "stiffens" the convergence approach path of the SUAV toward the desired range. The "stiffening" effect occurs as the control law increases emphasis in driving the navigation angle error, η to zero, while converging towards the desired range.

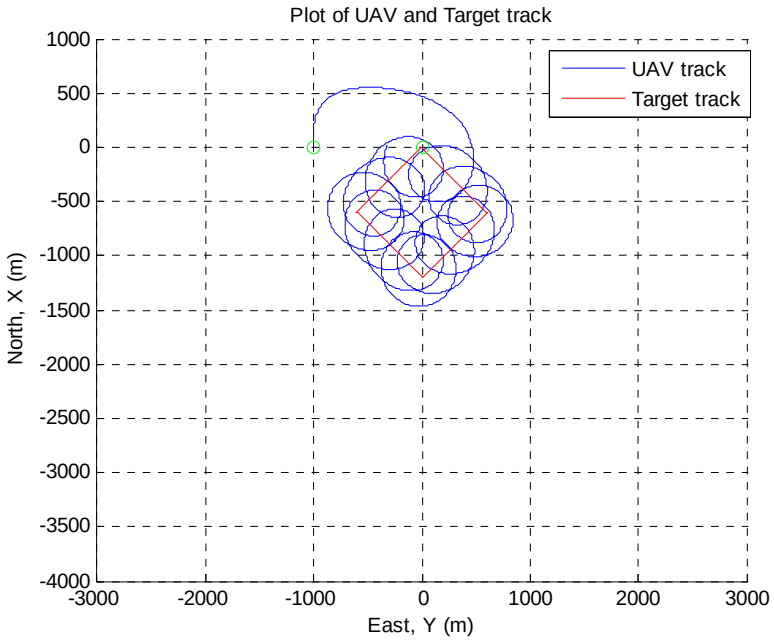
2. Sensitivity Analysis of the Control Law to Variations of *Vt/Vg*

The sensitivity analysis of the control law to variations of parameter *Vt/Vg* is examined in a scenario in which the desired range is 300m. The initial conditions are (1) SUAV velocity = 28m/s, (2) Initial position of SUAV is at [0, -1000, 300], (3) Initial position of target is at [0, 0, 0], (The initial horizontal ground range from UAV to target is 1000m), (4) K_1= 0.2, and (5) k_2 = 0.25.

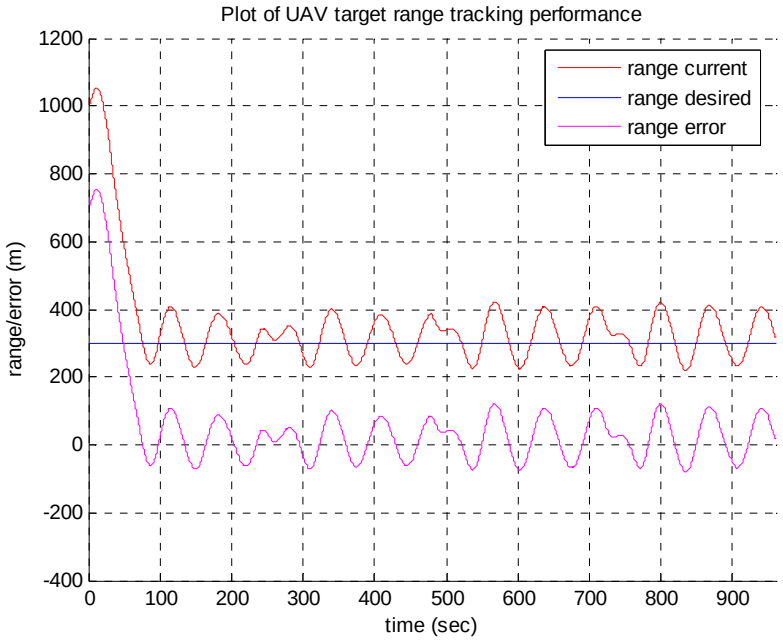
The previously defined MOPs will be utilized to assess the performance sensitivity of the newly developed control law to parameter V_t/V_g variations.

Case 1: Vt/Vg = 5/28

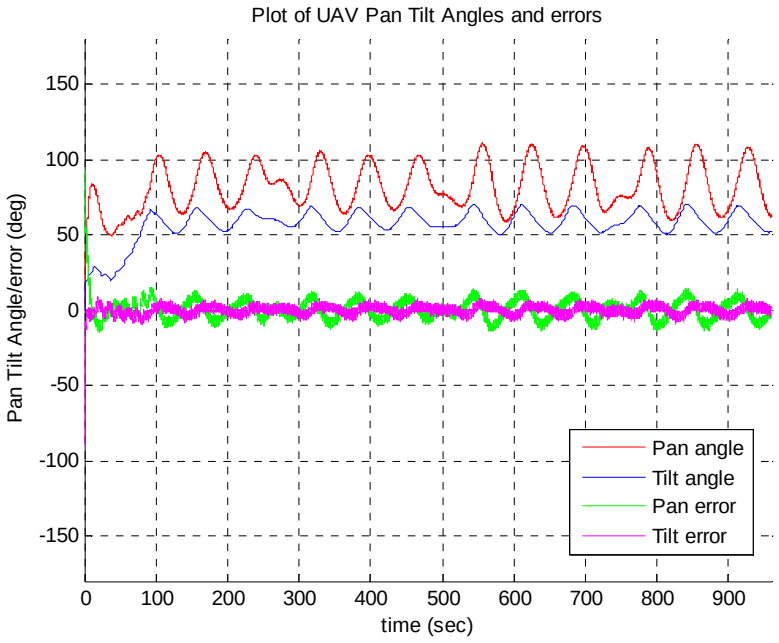
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 300m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

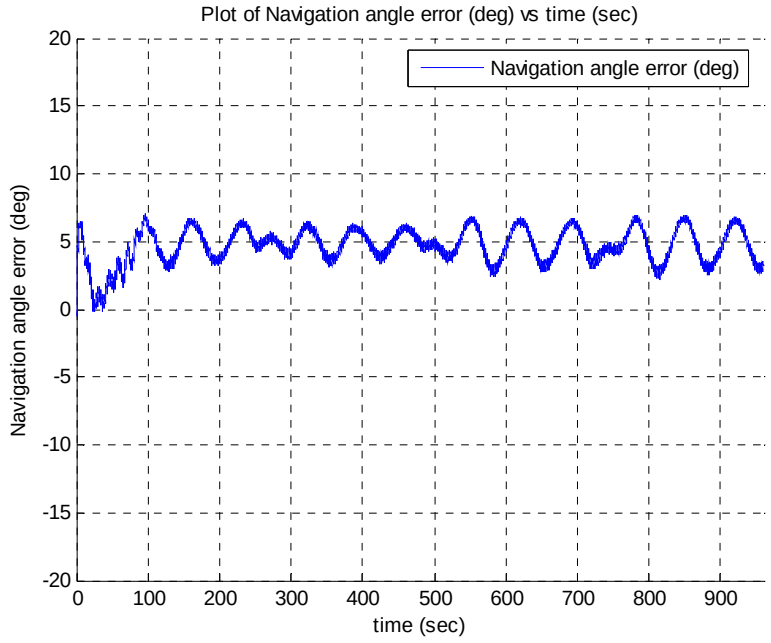
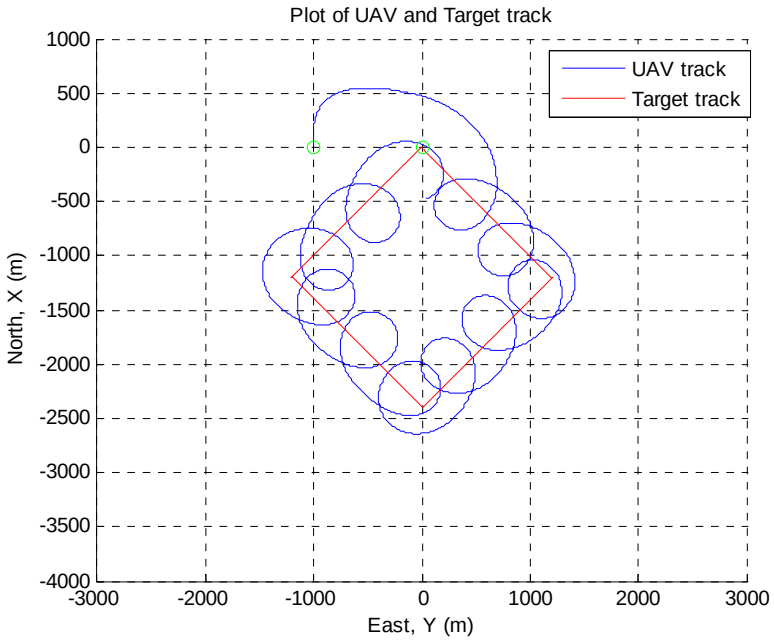


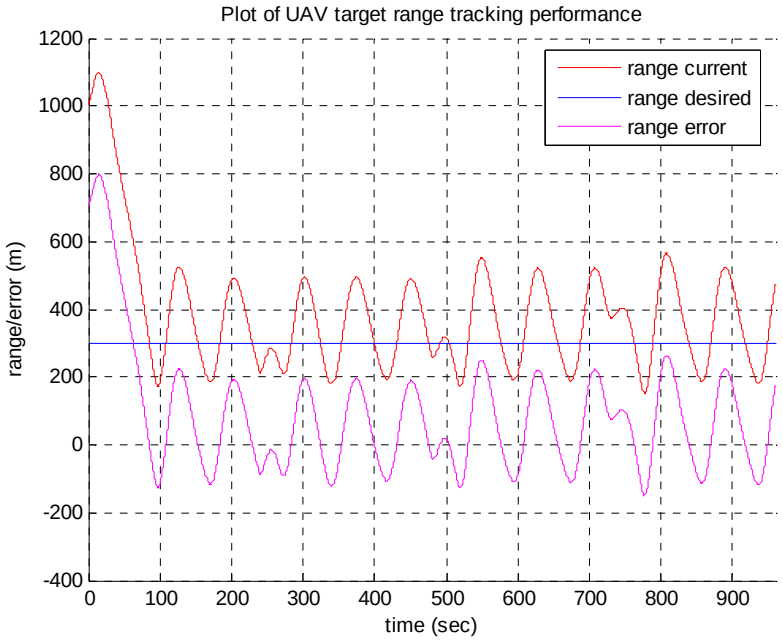
Figure 16: Sensitivity Analysis for Vt/Vg = 5/28 at Desired Range 300m

Case 2: Vt/Vg = 10/28

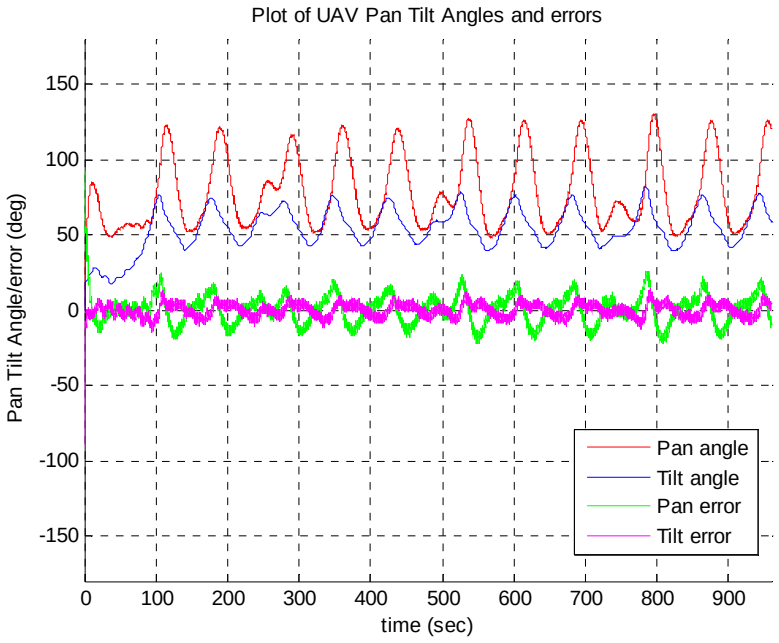
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 300m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

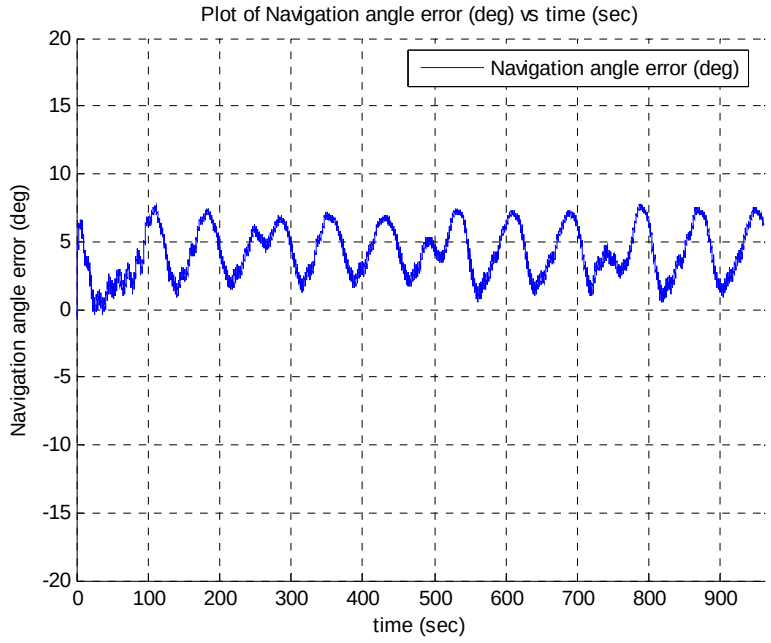
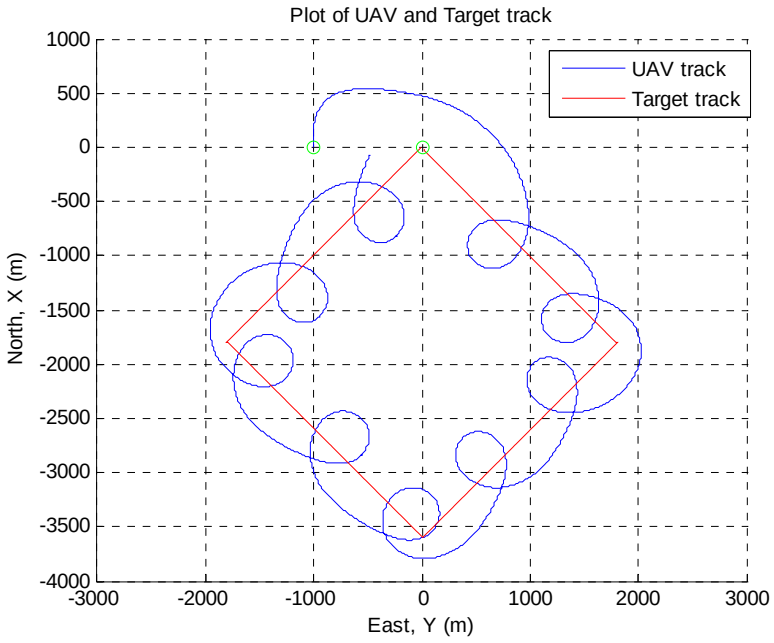


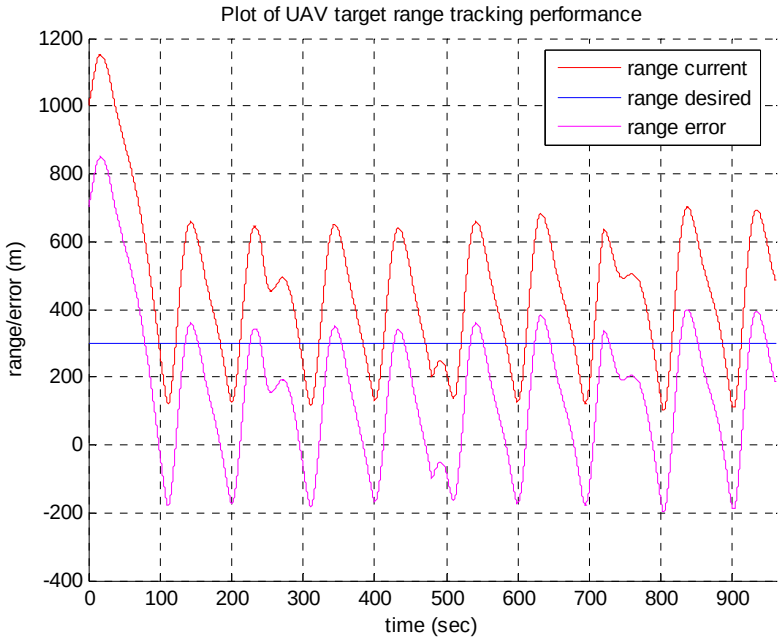
Figure 17: Sensitivity Analysis for Vt/Vg = 10/28 at Desired Range 300m

Case 3: Vt/Vg = 15/28

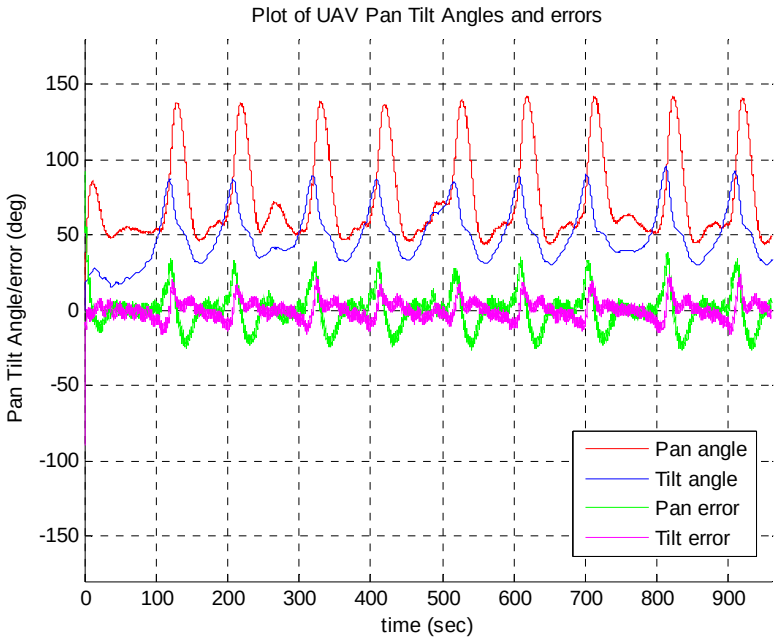
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 300m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

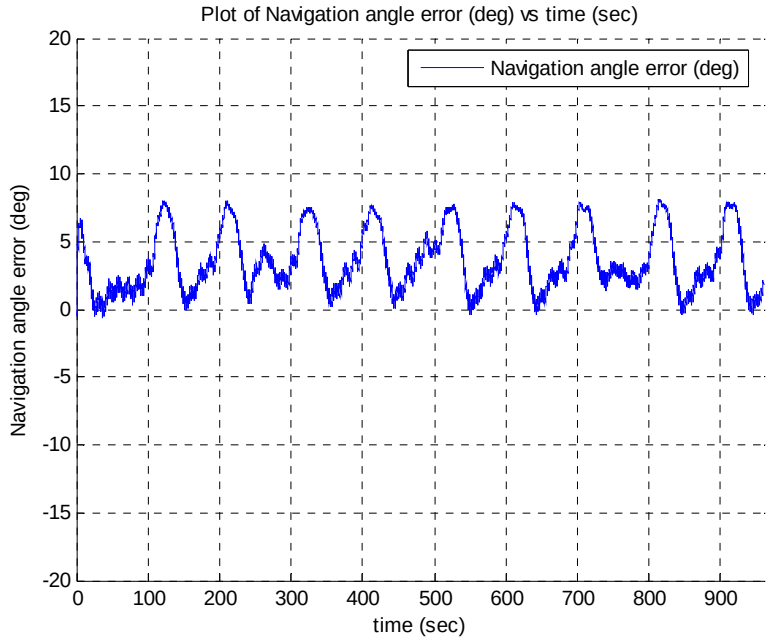


Figure 18: Sensitivity Analysis for Vt/Vg = 15/28 at Desired Range 300m

The contributors of MOP-1(Vt/Vg) for the 3 test cases are tabulated below:

Vt/Vg	Range Captured (m)	Time to first closest approach (sec)	MOP-1(k_1), convergence speed (m/sec)
5/28	700	80	8.750
10/28	700	90	7.778
15/28	700	100	7.000

Table 3: MOP-1(Vt/Vg)

The MOP-1(Vt/Vg) of the SUAV is plotted as follows:

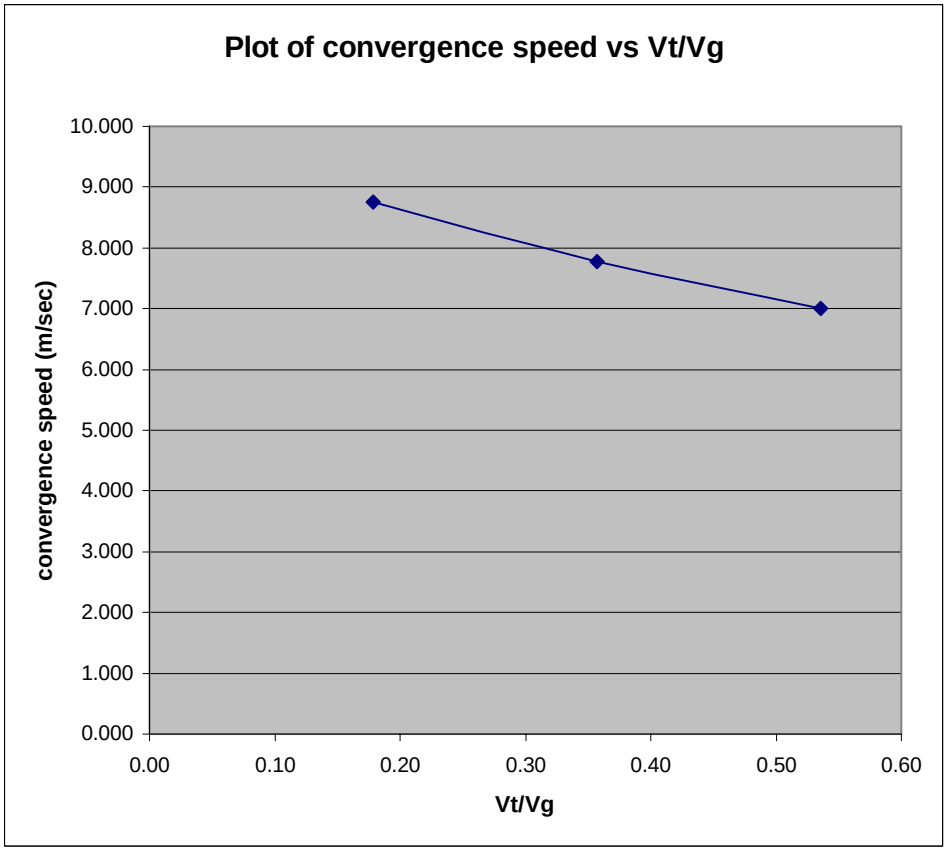


Figure 19: MOP-1(Vt/Vg)

The contributors of MOP-2(Vt/Vg) for the three test cases are tabulated below:

Vt/Vg	Maximum range deviation error (m)	MOP-2(k_1), Percentage deviation from desired range (%)
5/28	100	33.3
10/28	220	73.3
15/28	400	133.3

Table 4: MOP-2(Vt/Vg)

The MOP-2(Vt/Vg) of the SUAV is plotted as follows:

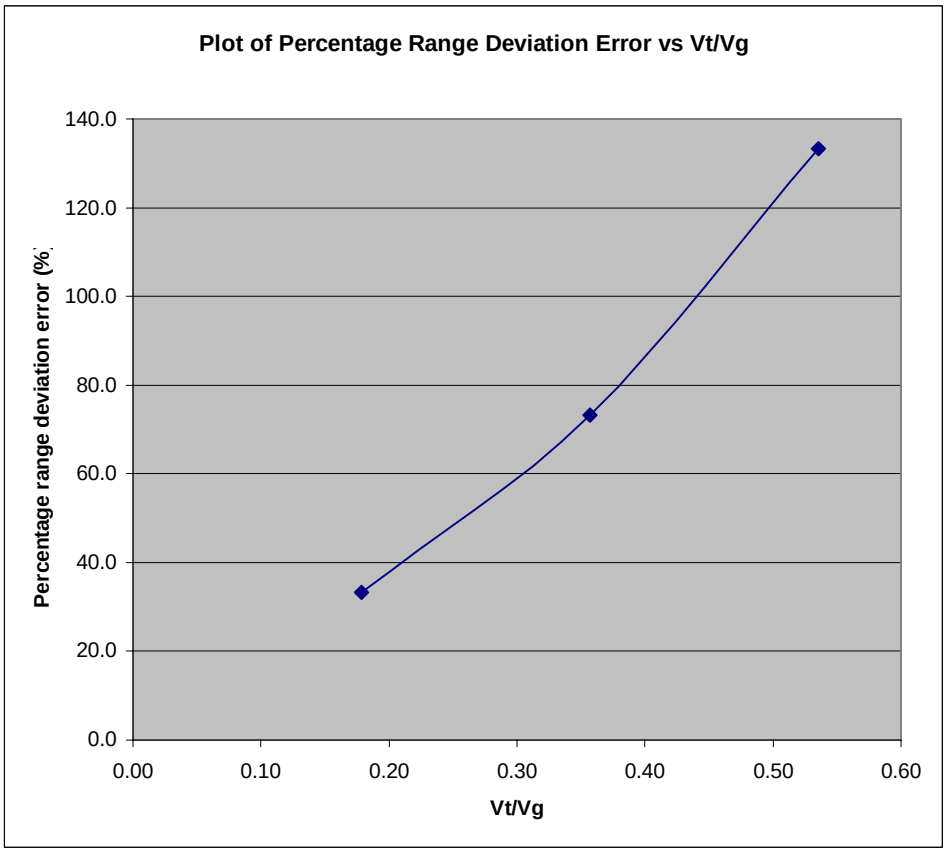


Figure 20: MOP-2(Vt/Vg)

The following observations can be made from the MOP plots above:

(a) From MOP-1(Vt/Vg) analysis, it can be observed that the convergence speed of the SUAV decreases with increasing Vt/Vg. The highest value is 8.75 (m/sec) when Vt/Vg is 5/28, and the lowest value is 7.00 (m/sec) when Vt/Vg is 15/28.

(b) From MOP-2(Vt/Vg) analysis, it is apparent that the percentage range deviation error of the SUAV increases with increasing Vt/Vg; the maximum range deviation error is 133.3% when Vt/Vg is 15/28, and the minimum is 33.3% when Vt/Vg is 5/28. The range holding performance of the new control law decreases rapidly with increasing Vt/Vg.

(c) From the test plots (Fig. 16 – 18) above, it is observable that the SUAV's range holding at the desired range decreases as the target velocity increases. This is intuitive, as the ability of the SUAV to turn depends on the velocity of the SUAV relative to the target. As the target velocity increases, the magnitude of the relative velocity decreases, thus diminishing the SUAV's ability for relative turn. This is evidenced by the decreasing number of circular approaches the SUAV can make against a target with a higher velocity.

(d) The ability of the SUAV to track through changes in target motion direction also demonstrates the robustness of the control algorithm in its adaptation to changes in target dynamics.

C. SIMULATION CONCLUSIONS

The SIMULINK results verify that the newly developed control law for the VBTT system operates in accordance to the theoretical predictions and has the following characteristics:

(a) Increases in k_1 effect a "stiffening" of the SUAV's approach path towards its desired range. The convergence speed of the SUAV decreases and levels off to a plateau with increasing k_1. The range holding performance of the SUAV, however, decreases with increasing k_1, after a minimum value.

(b) Both the speed of convergence and range holding performance of the SUAV degrade with increasing Vt/Vg.

D. HARDWARE IN THE LOOP SIMULATION

The next logical goal is to perform Hardware-In-The-Loop (HIL) lab testing of the newly developed control law for the purpose of better knowledge retention, continuation and proliferation of the VBTT system, and as part of the thesis secondary objectives. The processes and procedures involved in HIL setup and testing of the newly developed control law are examined and documented categorically. The HIL setup schematic is pictured in Figure 21:

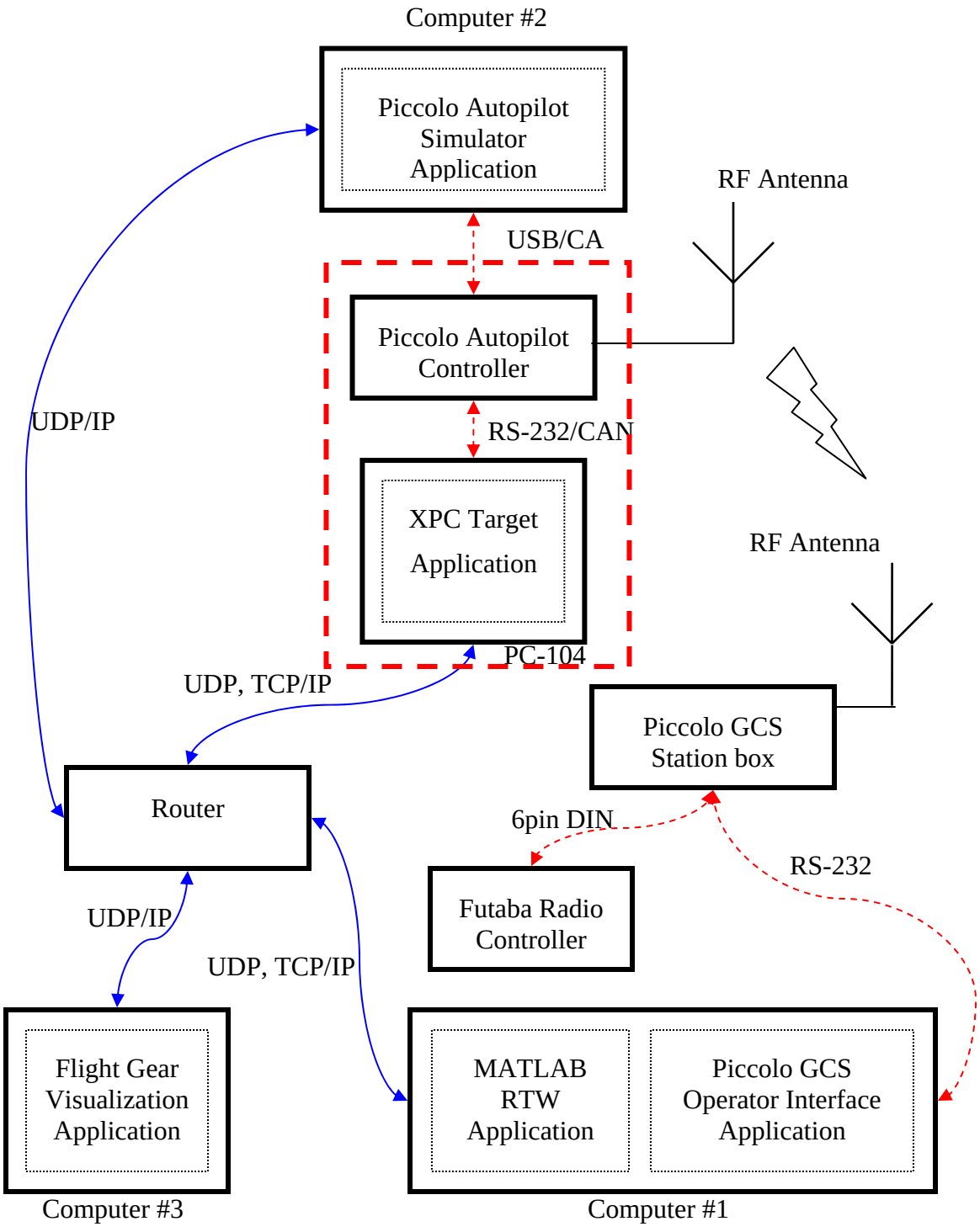


Figure 21: Hardware in the Loop Setup Schematic

1. Piccolo Autopilot Setup and Familiarization

Piccolo is a complete, integrated avionics system developed by Cloud Cap technology for small unmanned aircraft. The Piccolo control system consists of four main parts: an avionics control system that is mounted onboard the SUAV, a Ground Station, a computer for operator interface and mission monitoring, and a pilot manual control interface via a modified Futaba radio controller.

The Piccolo system employs two separate control loops; the faster inner loop controls the aircraft dynamics within the aircraft itself, while the slower outer loop controls the path that the aircraft is expected to follow via a wireless communication link between the piccolo autopilot controller and the ground control station. The inner control loop will be utilizing the control law developed earlier in this thesis to direct the flight control of the SUAV. Typical Piccolo HIL equipment and the system setup are shown as follows (Fig. 22). (All the information pertaining to Piccolo AP and Ground Station is taken from Ref 2, Ref 6 and Ref 8.)

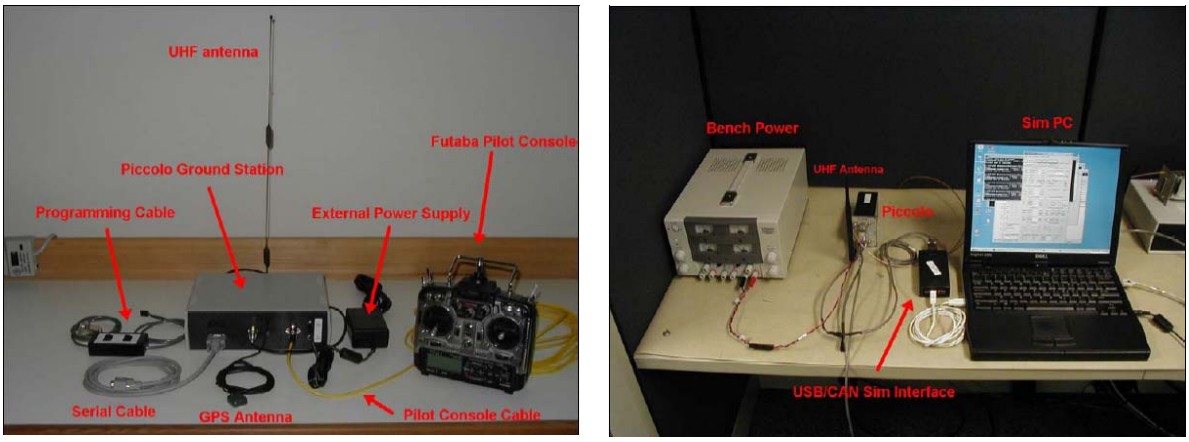


Figure 22: Ground Station setup and the SIM PC and avionics side of HIL Simulation

a. Piccolo Autopilot Controller Familiarization

The CPU of Piccolo AP is the MPC555 microcontroller, a new breed of automotive controller based on the PowerPC architecture, and capable of delivering 40 MHz PowerPC operation, including the hardware floating point.

Integrated within the avionic controller unit are three ADXRS300 gyros and two two-axis ADXL210e accelerometers. The Motorola M12 GPS provides Piccolo with its basic groundspeed and position. Included with the Piccolo interface are a dual ported mpxv50045 4kPa dynamic pressure sensor, an absolute ported mpx4115a barometric pressure sensor, and a board temperature sensor. A sophisticated data link, built on the MHX 910/2400 radio modem from Microhard Systems Inc, provides up to 40Kbaud of throughput and is used for command and control, autopilot telemetry, payload data transfer functions, and differential GPS corrections. The frontal panel included the filtered 44-pin vehicle interface connector, GPS and UHF antenna SMA connectors, and the Pitot and Static pressure port nipples. The Piccolo block diagram and front panel schematic is shown below (Fig. 23):

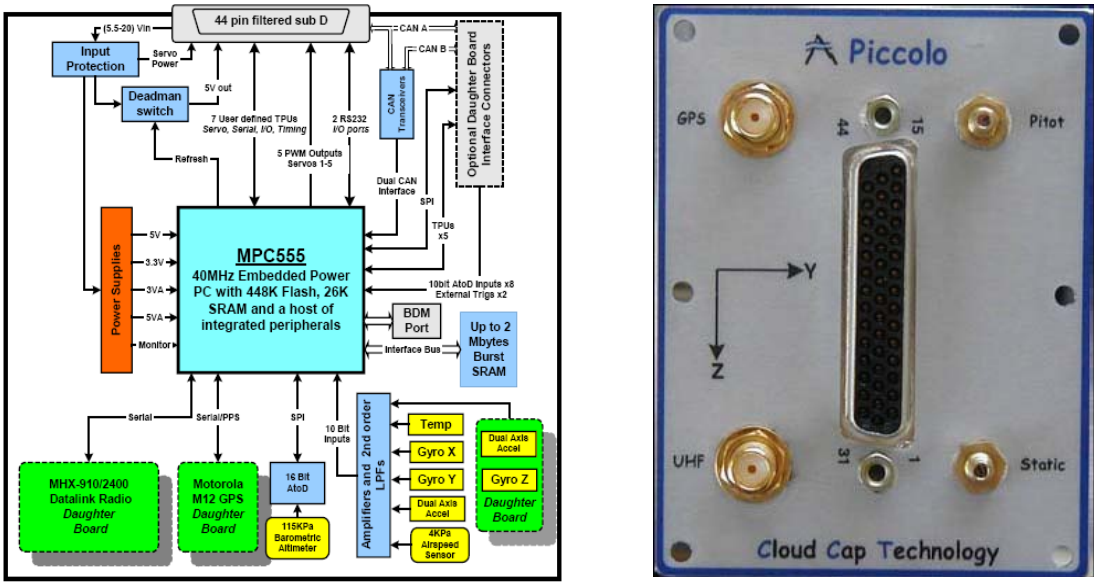


Figure 23: Piccolo Block Diagram and Front Panel

In the HIL setup and simulation environment, the Piccolo AP is connected to computer#2, which is running Simulator software. The inter-link between computer#2 and the Piccolo AP controller functions via a USB-CAN data exchange cable. The Simulator allows for testing of the aircraft control laws and mission functionality without jeopardizing the hardware in flight test. Piccolo's HIL simulator is based upon the external CAN interface; servo control information is sent and external sensor data

received over the CAN bus. Since the simulator communicates with the avionics in real time, it must itself run as a real time application; no other applications should be run on the PC (Computer #2) that runs the simulator.

After the Simulator program is launched, the aircraft model must be initialized. This is accomplished under the FILE menu options. A default "cub aircraft" model file is provided by the software, and its flight characteristics resemble that of the actual Tele-Master SUAV used for the flight test. Next, the start state of the Simulator is also initialized. A default file containing the geodetic data around the Camp Robert airfield test site is available in the system for quick initialization. The "Reset", "Apply slew", and "clear slew" radio buttons apply the typed customized parameters to the Simulator program. The Simulator program is started and stopped using the "Start" and "Stop" radio buttons. Turbulence parameters can be entered into the program to simulate weather and wind conditions. The Simulator program interface layout is shown in the figure below (Fig. 24):

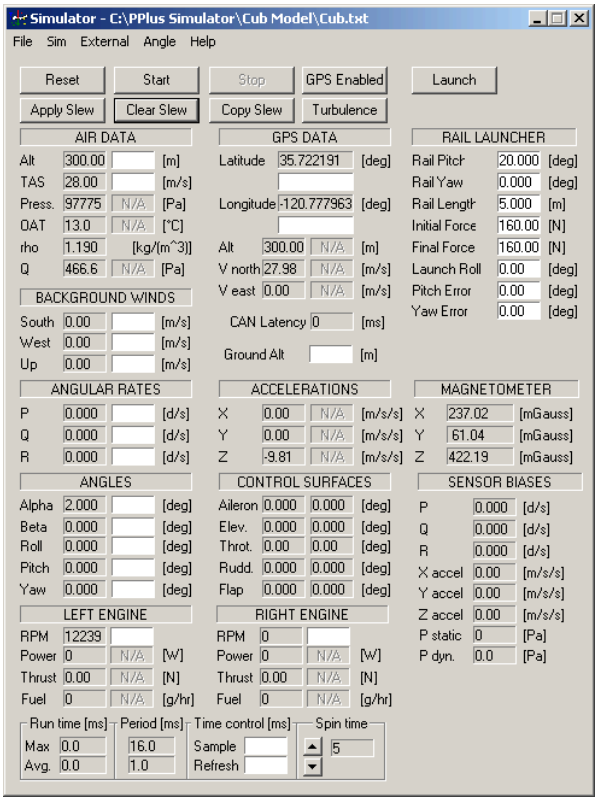


Figure 24: Simulator Program Interface

b. Piccolo Ground Control Station Familiarization

Piccolo's ground control station is based upon the same hardware that makes up the avionics package. The Ground Station manages the communication link to one or more avionics systems, interfaces to the pilot in the loop console, and provides a command and control stream to the operator interface PC. The station connects to the operator interface PC through a standard nine-pin serial cable. The Ground Station GPS Antenna is connected to the rear panel SMB connector, and the UHF antenna is hooked up to the BNC connector. A six-pin circular DIN pilot console cable connects the pilot in the loop command to the Ground Station through a modified Futaba radio controller. The Ground Station exchanges data with the Piccolo avionics via a built-in 900 MHz or 2.4 GHz ISM band radio wireless link, which is made from the MHX-910/2400 frequency hopping radio developed by Microhard Systems Inc. The Piccolo Ground Station is shown as follows (Fig. 25):

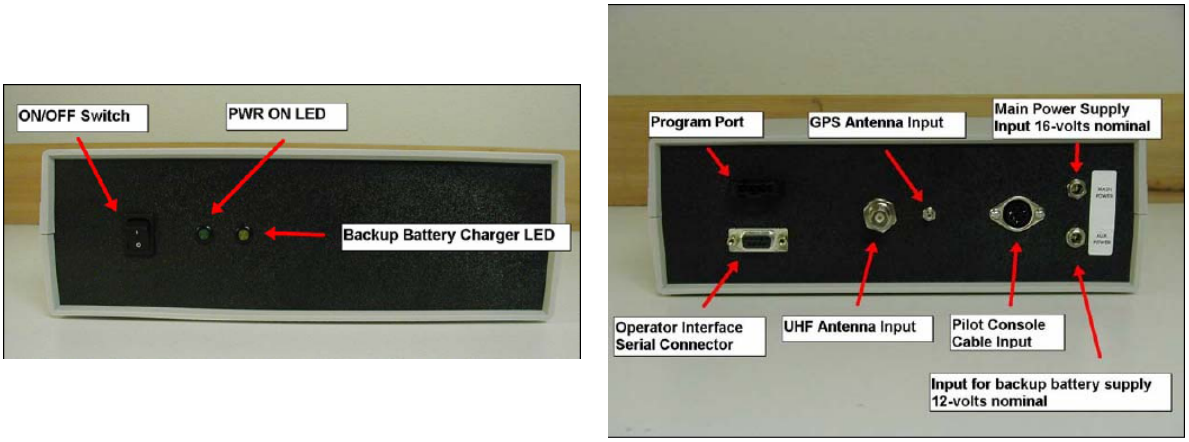


Figure 25: Piccolo Ground Station, Showing Front and Back Panels

The operator interface (OI) is a software system that runs on a Windows PC and provides a command and control interface for Piccolo operators. The operator interface communicates to the Ground Station over a RS-232 serial link (default to COM1). Installation of the OI is through a windows installer file, "Operator Interface.msi", which is downloadable from the Cloud Cap Technology site. The OI is installed in computer#1.

The OI provides two station screens for operator interface with the Ground Station. The Ground Station screen provides a Window Menu and Unit Menu so that the user may manage the avionic window display layout and units of telemetry that appear on the screen. Advanced options under the Window Menu display a more comprehensive version of the avionics window for operator interface. The Avionics Window displays telemetry data received from Piccolo avionics. A screenshot of the Piccolo OI is displayed in Figure 26:

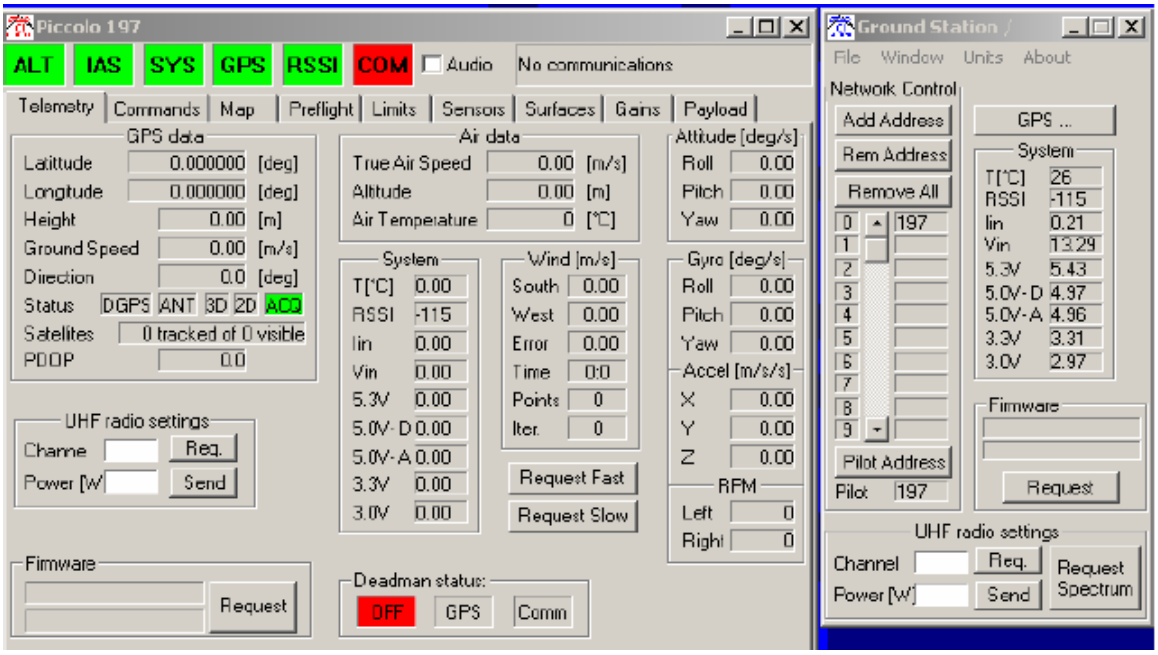


Figure 26: Screenshot of Piccolo Operator Interface

Essential telemetry information is displayed in tabulated pages selectable by the user. The tabulated pages are (1) Telemetry, (2) Commands, (3) Map, (4) Preflight, (5) Limits, (6) Sensors, (7) Surfaces, (8) Gains, (9) Payload, and (10) Parameters. Some essential pages compulsory in HIL setup will be briefly addressed as follows:

Telemetry page—The telemetry page displays data from all the sensors within the system. Sensor information chosen for display includes GPS data, Air Data, Sonic altimeter, MHX radio settings, System Version, Diagnostic, Wind, Attitude, Gyro,

RPM, and Compass. The "request fast" and "request slow" buttons alter the rate at which the telemetry data is sent. "Default slow mode" sends data once per second; "fast mode" sends data 20 times a second, or according to the limitations of available bandwidth.

AP commands—The autopilot commands page displays the current autopilot command status, and allows the user to change IAS, Altitude, turn rate, flaps, and waypoint tracker via an interactive input interface. Other commands, such as autopilot mode, stick mode, engine control, and flight actions, are selectable through on-screen radio buttons.

MAP—The map page displays the current location of the vehicle, and provides an interface for creating flight plans. The map is built on ESRI's MapObjects, and thus is capable of displaying geo-referenced raster files, as well as vector shape files. Customized maps in tif format can be added to the display using the "Add image layer" or "Add vector layer" commands.

AP limits—The limits page is used to view and alter the autopilot and mission limits. Autopilot limits include min/max of airspeed, altitude, bank angle, aileron, elevator, rudder, throttle, and flap, and can be input via an interactive interface. A "request and send limit radio" button is used to poll for and alter the current limits in the Piccolo avionics.[10]

Sensors—The sensors page gives the current sensor readings and calibration information for each sensor. The page is normally only used during the setup process. An important step in the HIL setup is to check and align the Euler angle axis of Piccolo avionics with the Platform Euler angle axis, out of 24 different possible axes orientations.

[10] Detailed procedures on setting of AP limits may be found at the Cloud Cap Technology web site: (http://www.cloudcaptech.com/download/Piccolo/Version%201.3.2/Docs/Piccolo%)-p. 41, 4.4.3.1)

AP Gains and Trims—The gains page is used to view and alter the autopilot gains and trims. There are gains for eight loops. Before the Piccolo AP can be profitably utilized for HIL simulation and control, the gains of the AP aircraft model must be tuned correctly. [11]

c. Flight Gear Visualization Environment Familiarization

During HIL simulation, it is important to be able to visualize the effects of the control law on the flight performance and stability of the aircraft. An open source application, Flight Gear, is provided for the visualization of aircraft attitude through a UDP/IP network protocol. In order to get Flight Gear to accept the state packet, the user must start with the correct command line switches. The batch file "runflightgear.bat" is provided for this. The Flight Gear application is installed on computer#3, equipped with a video card designed for OpenGL hardware acceleration. The flight visualization is activated from Piccolo Simulator Flight Gear output interface selection (Computer #2) by specifying the IP address/Name of computer#3. A screen shot display of the Flight Gear visualization is shown in the following figure (Fig. 27):

Figure 27: Screen Shot Display of Flight Gear Visualization

[11] Detailed procedures on tuning of AP gains and trims are found at the Cloud Cap Technology web site: (http://www.cloudcaptech.com/download/Piccolo/Version%201.3.2/Docs/Piccolo% (pp. 41 – 43, 4.4.3.2/4.4.3.3)

2. PC104 Setup and Familiarization

The PC/104 computer (Fig. 28) is a stack of small 3.5" square cards connected by an ISA communication bus. It is a popular standardized form-factor for small computing modules typically used in industrial control systems or vehicles. The PC/104 bus includes CPUs, video controllers, net-work interfaces, sound I/O, data acquisition boards, serial RS-232, and specialized interfaces. The PC/104 stack downloads the XPC Target model from the Host PC via a TCP/IP network. (All the Information pertaining to PC104 is extracted from Ref 9 and 10)

Figure 28: PC104

Before the PC/104 can accept the compiled XPC Target model from MATLAB/RealTimeWorkshop in the HIL simulation, it must be properly setup. A detailed step by step instruction/reference on the basic set up of the PC/104 and its peripherals is available at Ref 9 and 10. Once the PC104 is correctly configured, it is connected via the UDP, TCP/IP network to accept compiled the XPC Target model from the SIMULINK RTW application on computer #1. The inter-link between PC104 and Piccolo AP controller connects through a RS232-CAN bus cable for data exchange. For further data analysis and real-time graph plotting, the XPC Target model outputs can be similarly extracted via UDP/IP at near real-time to another SIMULINK Data Extraction Model.

3. HIL Network Communication, Data Exchange/ Collection and Familiarization

a. Router Network Communication Protocol

The HIL network communication setup uses two forms of network communication protocol; they are TCP/IP and UDP/IP. TCP/IP uploads the XPC Target model from Host PC to PC104. UDP/IP downloads the XPC Target model outputs from PC104 to Flight Gear's computer for visualization of the aircraft attitude and performance, as well as extraction of relevant XPC Target data to another SIMULINK Data Extraction program running on Host PC (Computer #1). The XPC Target data is downloaded for the purpose of post-simulation data analysis and real-time graph plotting.

TCP/IP provides reliability of data delivery and accuracy between the sending and receiving party, but at the expense of an incurring time delay when network traffic is heavy. UDP/IP, however, is connectionless and unreliable; the sending party will send out information whenever it can, and the receiver will receive information whenever it is able to do so. Information sent when the receiver is unavailable will simply be lost. For real-time applications, since only the most recent information is of relevance, UDP/IP is widely employed as the choice network communication protocol. The types of network communication protocols between the HIL computers and PC104 are depicted in Figure 26.

b. Setting Communication to Piccolo AP in HIL

The control architecture of modern autopilots is usually based on data fusion from numerous heterogeneous sensors and various control channels. As a result, an entire control system can be represented as a complex multi-rate process in full duplex mode. Therefore, one of the principal requirements to the onboard sensors consists in providing measurements of the process with an update-rate capable of representing the physical phenomenon. In turn, the actuator subsystem is also required to deliver control actions in order to affect the dynamics of the system.

Serial Interface (SI) communication technology is widely employed in modern control data interfacing for the purpose of establishing the link between the

ground control unit and the airborne vehicle. Mathwork's xPC target provides an RS232 library that supports a variety of serial instruments. These drivers support synchronous, asynchronous, and binary (asynchronous) communication modes. The latest xPC target release supports RS232/422/485 protocols and a number of serial communication boards, including Quatech and Diamond's products; however, the technology does not extend the reading capability of new serial data formats, such as proprietary communication protocol in Piccolo AP.

Mathwork's library directly supports an application of a binary asynchronous communication that relieves the user of the obligation to develop an actual RS232 hardware driver. The application conveniently allows the user to focus on the implementation of data interfacing rather than hardware programming. Since the majority of SI formatting is proprietary, the most efficient way to implement the SI capability consists in writing Level-2 S-functions. The technique used in this thesis provides the operational separation of principal functions among the pre-built library blocks and the user-developed S-functions. Standard serial communication blocks deliver raw binary data to the model using the optimized Mathworks' routines, and the S-functions perform format specific processing that is not readily available in Mathwork's library. See (Fig. 29) below for depiction of this concept. (Ref 11)

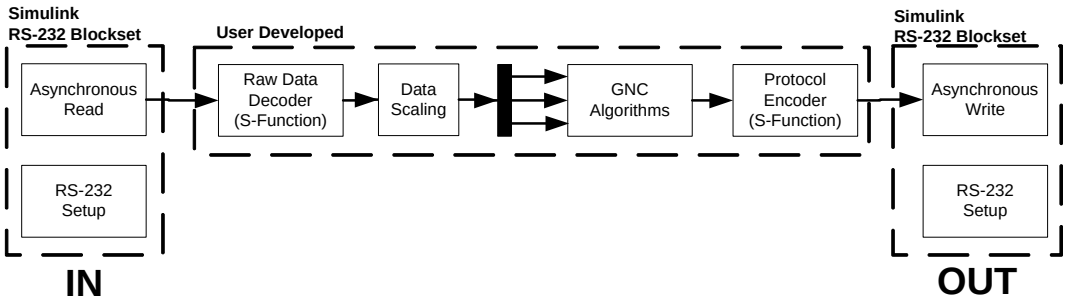


Figure 29: Separation of Interface Functions

A customized SIMULINK RTW communication program has been developed for the data exchange between the XPC Target model and Piccolo avionics. The model is shown below in Figure 30.

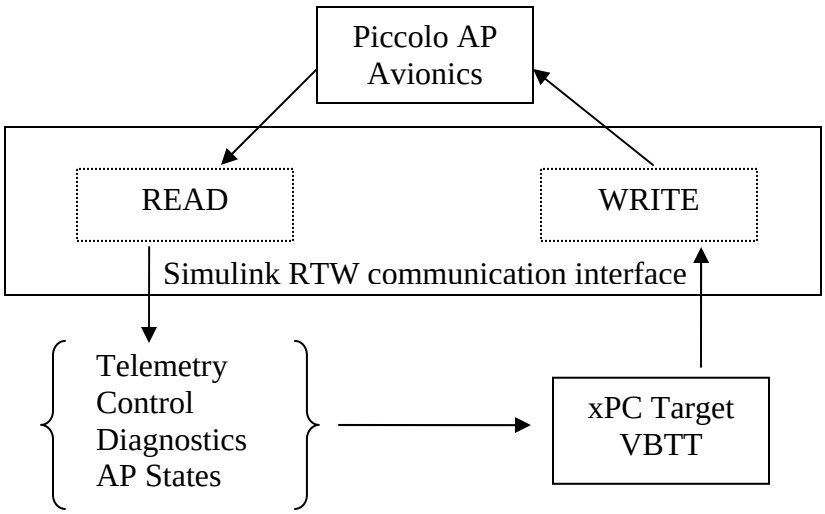


Figure 30: SIMULINK RTW Communication Interface

Using the customized SIMULINK RTW communication interface program, relevant telemetry, control, diagnostics, and AP-states information can be readily extracted from Piccolo avionics output for the XPC Target model guidance control computation. Subsequently, the computed control commands from the XPC Target model can be translated into Piccolo communication protocol format for control of Piccolo avionics.

The SIMULINK RTW communication interface program is comprised of MATLAB standard RS232 serial binary communication blocks and customized programmed Level-2 S–functions. The SIMULINK RTW communication interface program applies colored coded blocks and paths to represent the WRITE (GREEN) and READ (RED) blocks and paths between the XPC Target and Piccolo avionics. To ensure that the SIMULINK RTW communication block is communicating to the correct Piccolo avionics unit, the user must properly input the Piccolo avionics unit ID into the program.

Reading from Piccolo AP

The READ block of the SIMULINK RTW communication interface has, built into the system, five customized S-function blocks named “pplus_readstream.c”, “pplus_autopil.c”, “pplus_diag.c”, “pplus_control.c”, and “pplus_telemetry.c”. The input parameters for S-function “pplus_readstream.c” are the Piccolo Autopilot ID (APID) and

the buffer size (BUF). The output from the S-function comprised of "data" and "header type" is parsed through a decoder sub-block. The other four S-functions nested within the decoder sub-block are utilized to decode the proprietary Piccolo communication protocol in order to separate out the Piccolo' telemetry, control, diagnostics, and AP states information. The READ block of the RTW communication interface program is depicted in the figure (Figure31) below:

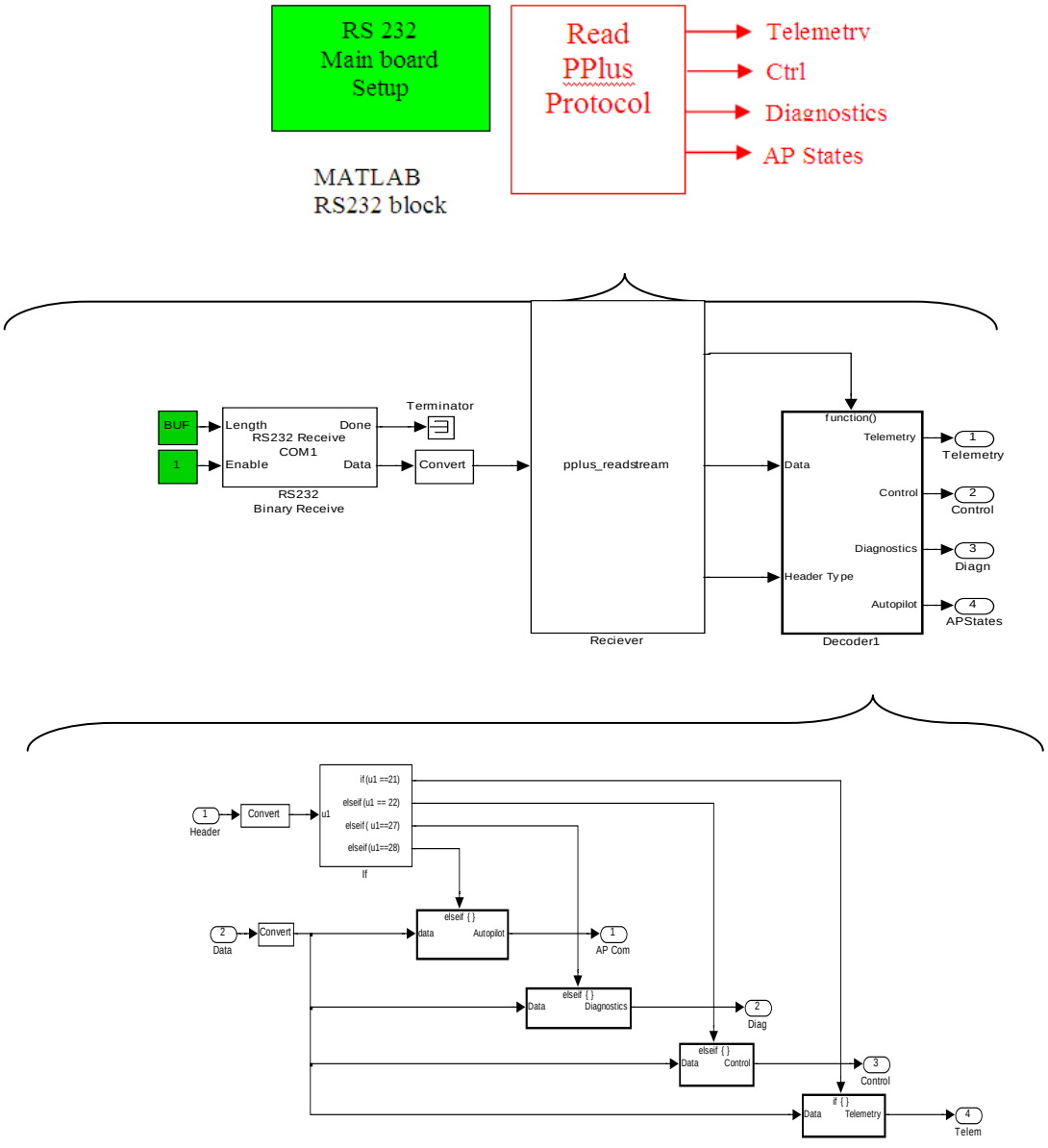


Figure 31: READ Block Of SIMULINK RTW Communication Interface Program

Writing to Piccolo AP

The WRITE block of the SIMULINK RTW communication interface program converts and codes the xPC VBTT guidance control message into proprietary Piccolo communication protocol before sending it through a standard MATLAB RS232 serial binary communication block.

A MATLAB sub-block converts the xPC Target guidance commands into "AP payload data stream". The input to the sub-block is the AP control loop number, control type, and control command values. The output from the sub-block is eight bytes of "AP loop message", also known as the "Payload" for the Piccolo AP.

An AP Stream Wrapper block codes the "AP loop message" or "Payload" into a proprietary Piccolo communication binary data stream. The proprietary Piccolo communication binary data stream is coded in a 2 layers communication protocol format. An outer layer discerns the type of data stream for "header" and "checksum" purposes, and an inner layer differentiates the type of command payload "packet" information. Three customized S-functions, namely "toplevelcrc.c", "enc-apilot_loop_fix.c", and "enc_top_level26.c", are nested within the WRITE block to perform the checksum and proprietary coding functions. The coded binary data stream is subsequently transmitted through a MATLAB standard RS232 serial communication block to Piccolo AP. A typical WRITE block of the RTW communication interface program is depicted in Figure 32, below:

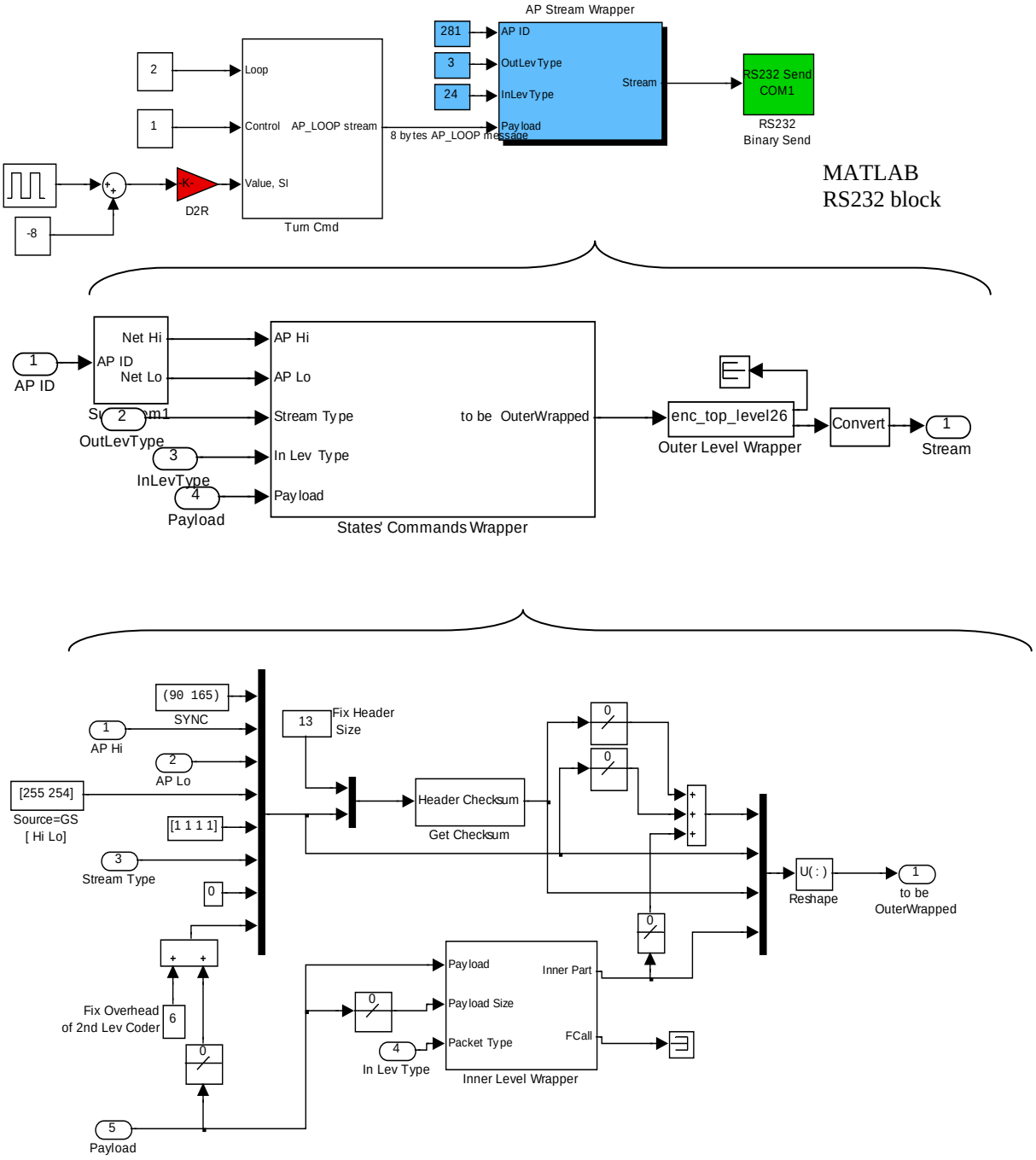


Figure 32: WRITE Block of SIMULINK RTW Communication Interface Program

A summary of the S-functions and their functions are tabulated as follows:

Where	Name of S-Functions	functions
READ Block	pplus_readstream.c	Extract messages from the Piccolo RAW data, mark output with a header
READ Block	pplus_autopil.c	Parse Piccolo AP states
READ Block	pplus_diag.c	Parse Piccolo diagnostic
READ Block	pplus_control.c	Parse Piccolo Control
READ Block	pplus_telemetry.c	Parse Piccolo Telemetry
WRITE Block	enc_top_level26.c	Outer Level Wrapper
WRITE Block	Toplevelcrc.c	Header / Checksum
WRITE Block	enc-apilot_loop_fix.c	Inner Level Wrapper

Table 5: Summary of S-Functions

An "INIT-GUI" M-script file initializes all the Piccolo AP state variable names and values used in the SIMULINK RTW communication interface program. The "INIT-GUI" M-script file is placed under the "Model Properties / Model callbacks" block for initialization.

A separate UDP communication SIMULINK program, utilizing the "Pack, UDP-Send-Binary" in SIMULINK library, is used to output the XPC Target output data stream (from PC104) to another SIMULINK Data Extraction program running on Computer #1. The SIMULINK Data Extraction program utilizes the "UDP-Receive-Binary, UnPack" UDP communication block to receive the data stream. This arrangement alleviates the computational constraints of PC104 in displaying the real-time data stream. The extracted XPC Target data is utilized for post simulation data analysis and real-time graph plotting utility for visualization of the simulation progress (On Computer #1).

4. MATLAB Real-Time Workshop/ xPC Target Setup and Familiarization

XPC Target is a real-time operating system based on the SIMULINK RealTimeWorkshop (RTW) environment. Programs are created through simple drag-and-drop procedures, and require very little "coding". SIMULINK models are compiled on a Host PC into C code programs that run on a Target PC. The system's "real-time" capabilities allow code to be executed at very regular intervals, which is optimal for control.

In order to create an xPC target model for HIL simulation on PC104, one must modify the model developed earlier for simulation in the SIMULINK environment. The AP controller block and the 6-DOF aircraft model block are replaced with the actual Piccolo AP controller hardware under a simulated flight control environment provided by the Simulator software. The Simulator software provides the sensor input necessary for the Piccolo AP controller to properly compute the appropriate simulated flight parameters of the aircraft.

a. XPC Target Model Schematic

The modified SIMULINK model schematic for xPC Target model compilation is depicted below in Figure 33:

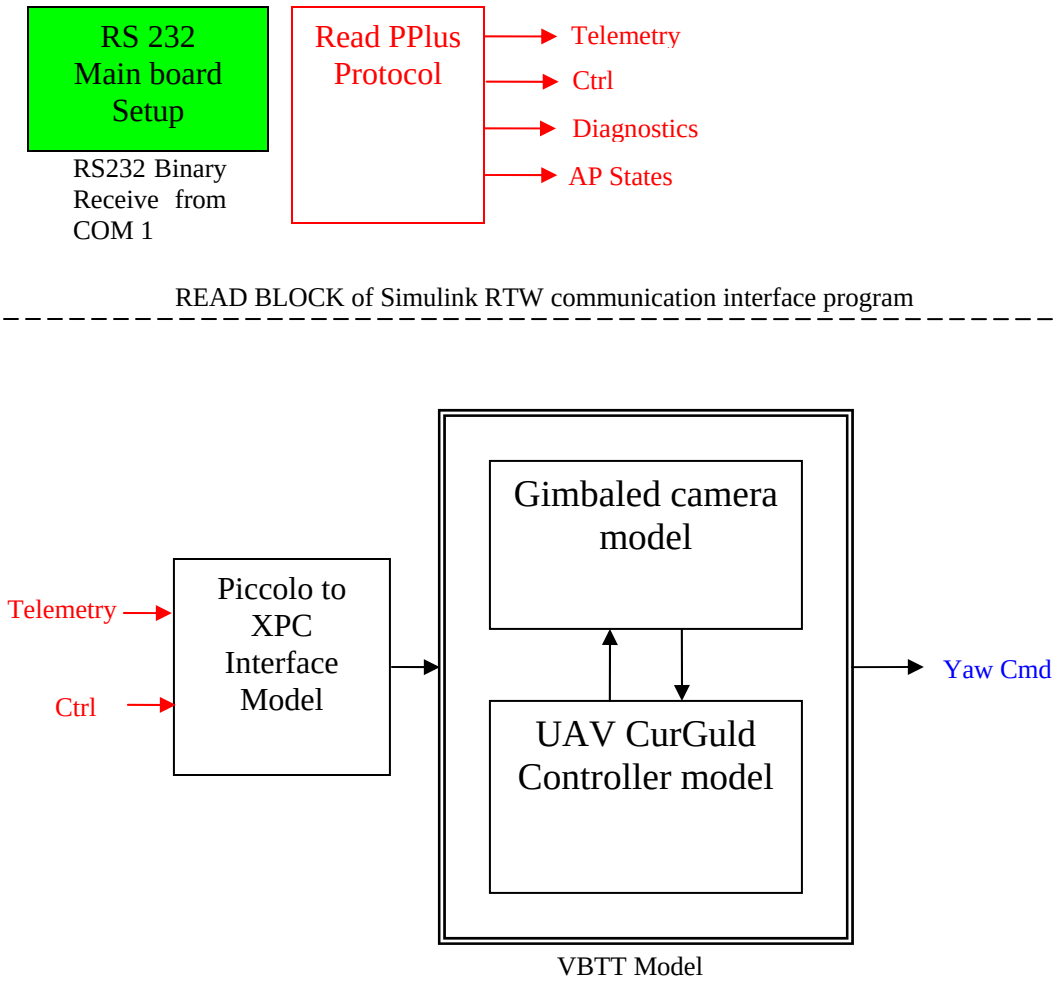


Figure 33: XPC Target Model Schematic

b. Code Building and Loading of XPC Target Model to PC104

In the next step of the configuration process, the user compiles the modified SIMULINK model on the Host PC (Computer #1) and downloads the compiled xPC Target Model to PC104. Before this can be accomplished, the "Configuration Parameter – Real-Time Workshop" window in SIMULINK must be configured appropriately so that SIMULINK may utilize the appropriate C language compiler. As xPC Target is a real-time system, the time step in the "Solver" option must be set to a fixed step instead of a variable step option.

Input via an *xPC Target Explorer* window, accomplished by typing "xpcexplr" in the MATLAB command window, allows the user to set up the communication path used to send the compiled C code file to the Target PC (PC104). The network IP address, TCP/IP gateway address, TCP/IP target driver, and port number of PC104 must be properly configured for the Host PC to communicate with the Target PC. The appropriate TCP/IP target driver is I82559 and port number is 22222 for the PC104 in use. The procedures are qualitatively depicted in the following figure (Fig. 34):

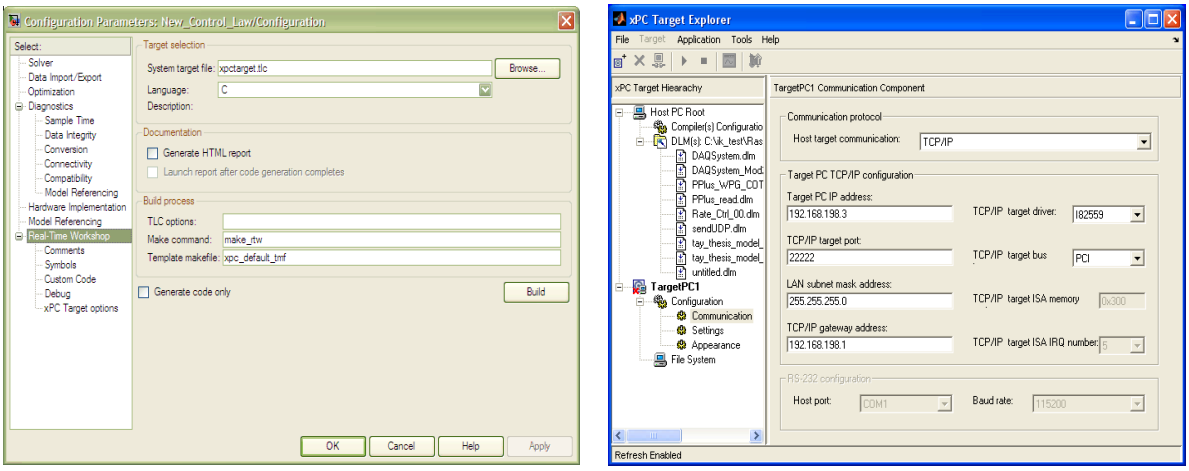


Figure 34: XPC Configuration Parameters and XPCEXPLR Window

Once all the configuration parameters are properly setup, generation of the xPC Target model is activated via a short cut key, "Ctrl-B". This initiates the automatic compilation process of the SIMULINK model into the C code program, and the compiled program will be automatically downloaded to PC104 via TCP/IP. It can then be triggered

to run externally using the SIMULINK External mode on the Host PC, or through xPC commands in the MATLAB command window or the M-script file.

E. HIL SIMULATION RESULTS

Utilizing the compiled XPC Target model and Piccolo avionics hardware, the new control law is tested in the HIL environment against the stationary and moving target test cases used in the earlier simulation.

One distinctive difference between software and HIL simulations is the presence of very noticeable levels of noise in the HIL simulation results. As there are more uncontrollable parameters and variables in the HIL simulation environment, there will be discrepancy between theoretical and actual HIL results. As such, results obtained through HIL should be interpreted in light of this unfortunate aspect. In an extremely noisy HIL environment, erroneous results might need to be filtered or selectively rejected for possible interpretation of results or analysis work.

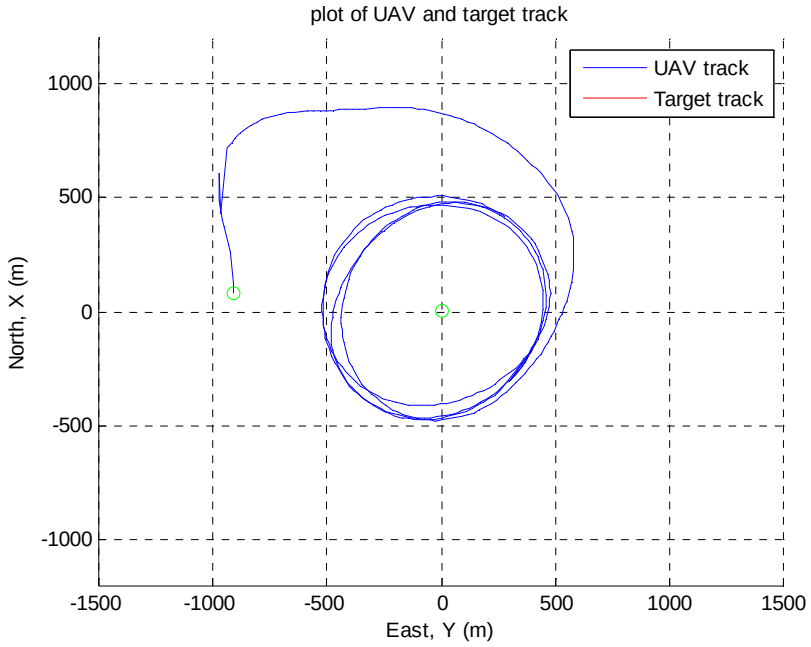
1. Sensitivity Analysis of the Control Law to Variations of K_1

The sensitivity analysis of the control law to variations of parameter k_1 is examined in a scenario in which the desired range is 500m. The initial conditions are (1) SUAV velocity = 28m/s, (2) Target velocity = 0 m/s, (3) Initial position of SUAV at [100, -900, 300], (4) Initial position of target at [0, 0, 0] (The initial horizontal ground range from UAV to target is 900m), and (5) $k_2 = 0.25$.

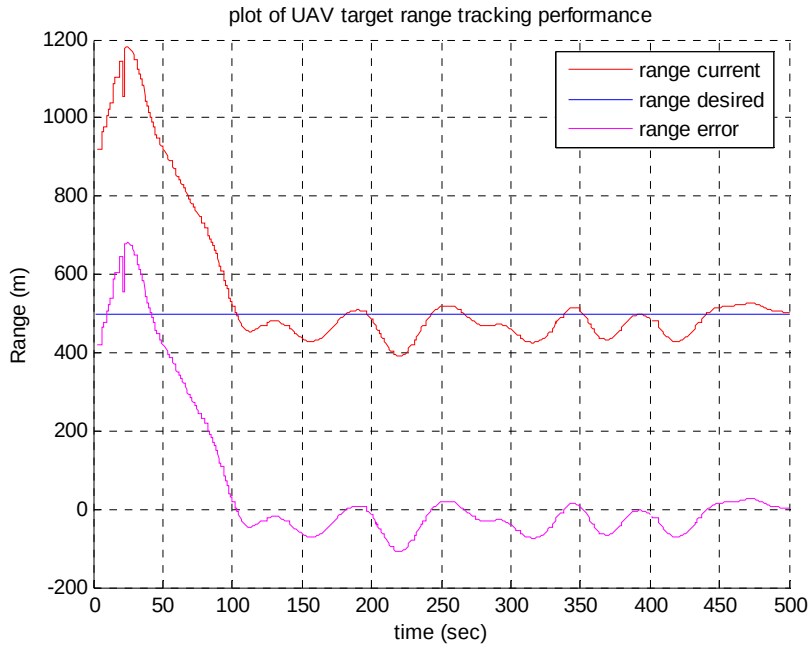
The same MOPs defined previously will be utilized to assess the performance sensitivity of the newly developed control law to parameter k_1 variations.

Case 1: K_1= 0.1

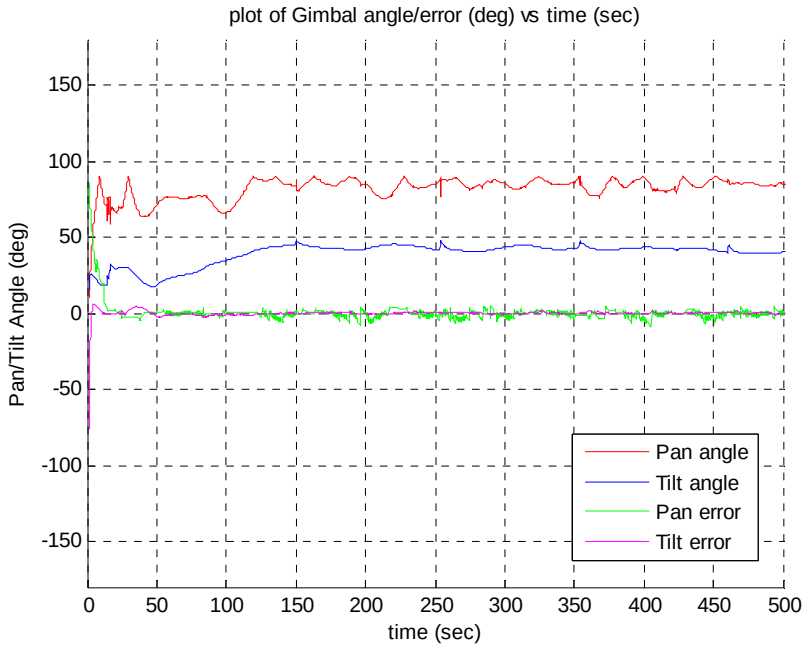
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 500m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

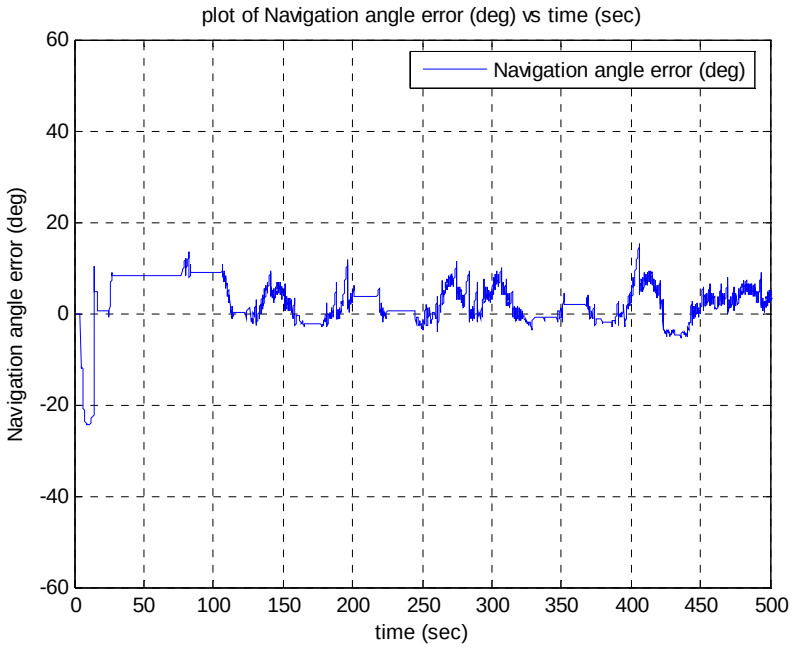
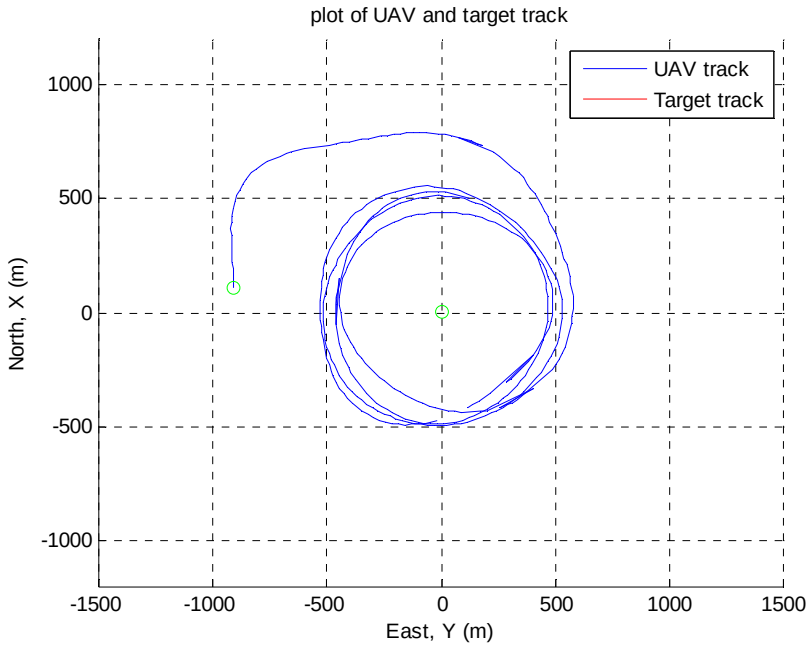


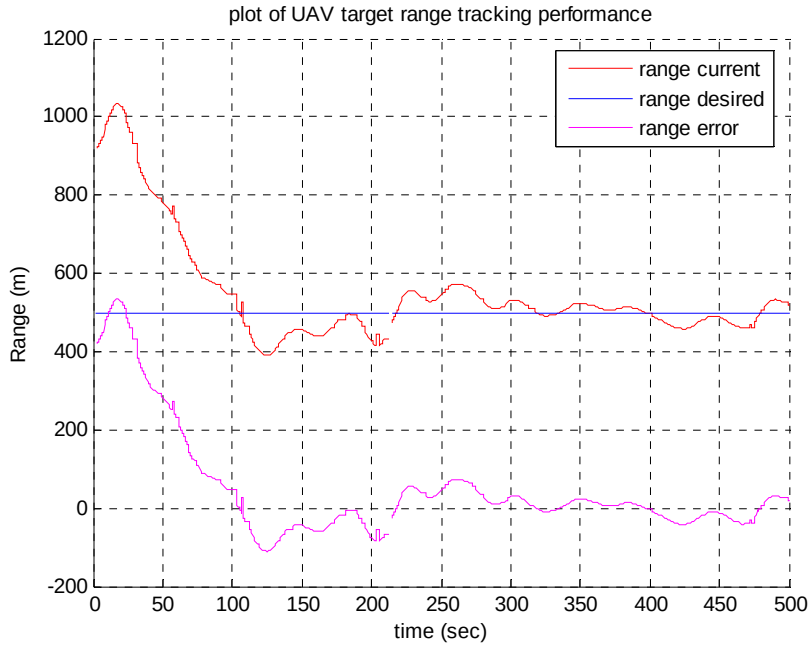
Figure 35: Sensitivity Analysis for K_1= 0.1 at Desired Range 500m (HIL)

Case 2: K_1= 0.2

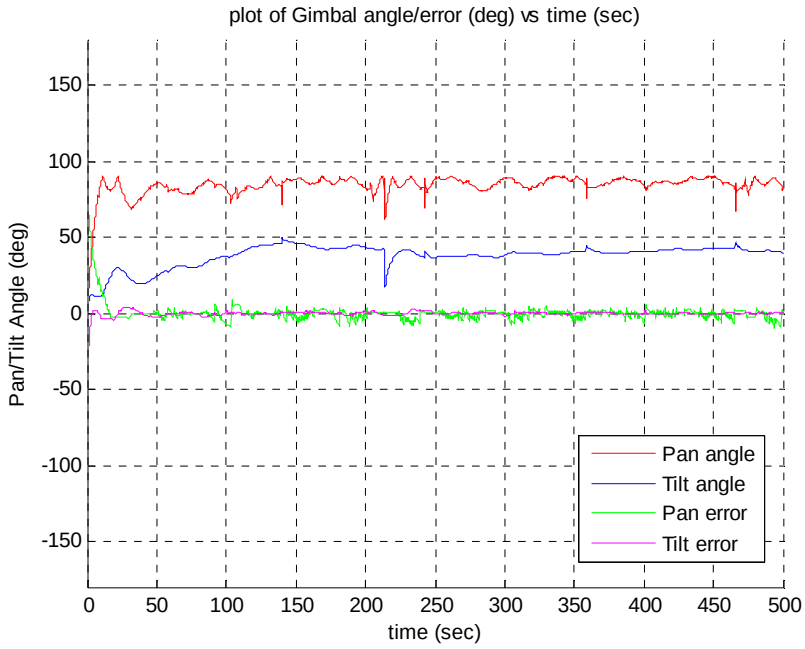
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 500m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

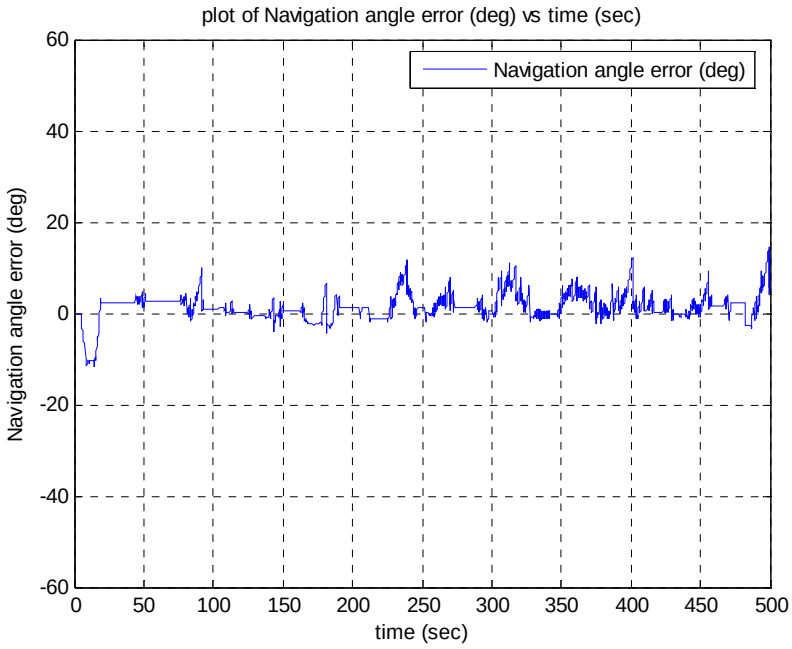
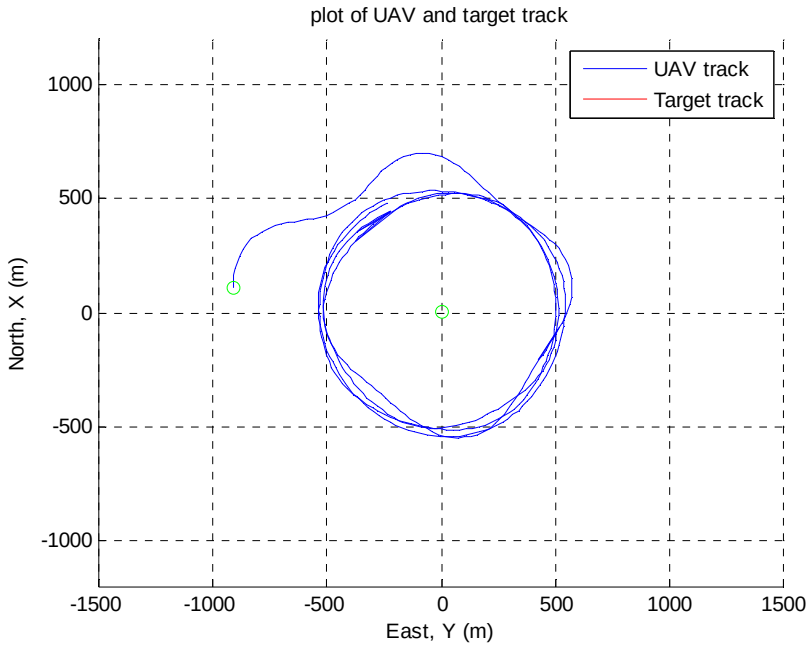


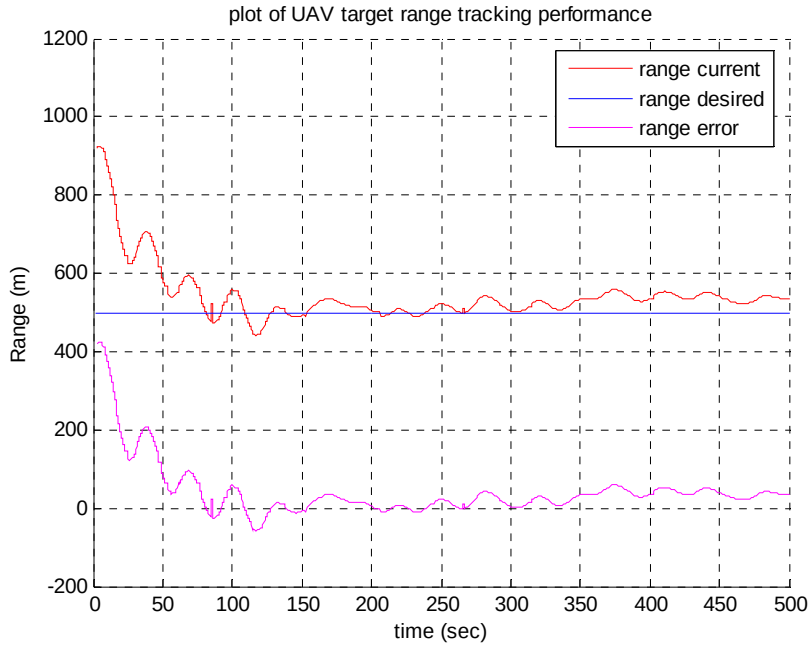
Figure 36: Sensitivity Analysis for K_1= 0.2 at Desired Range 500m (HIL)

Case 3: K_1= 0.3

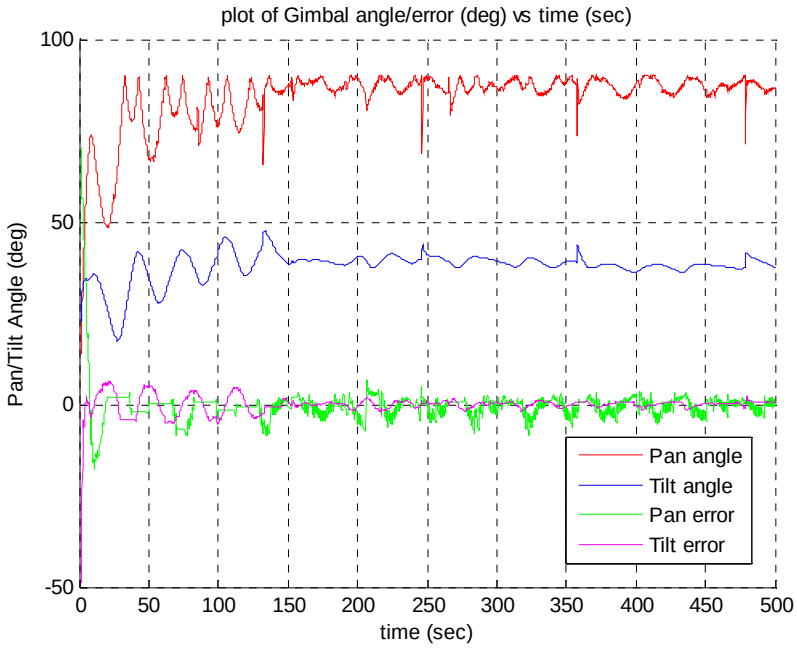
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 500m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

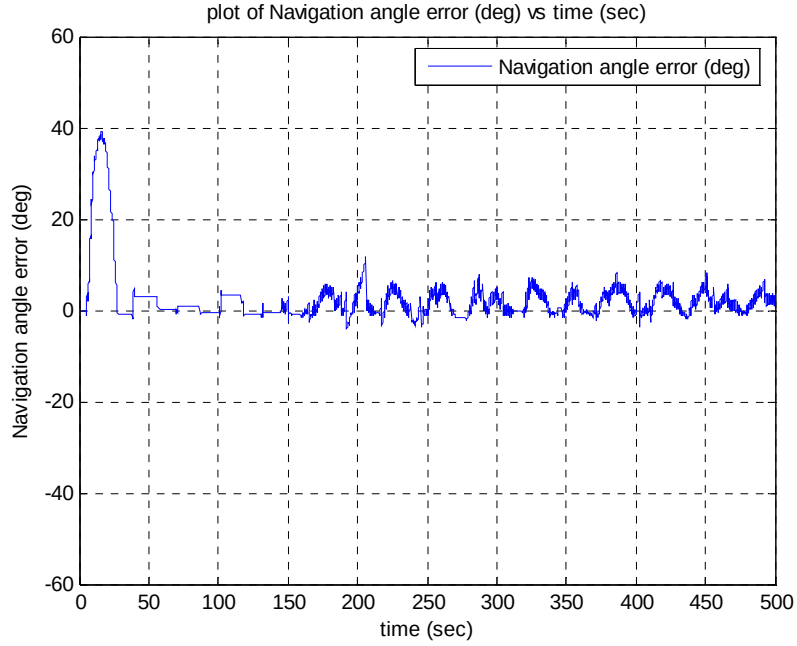
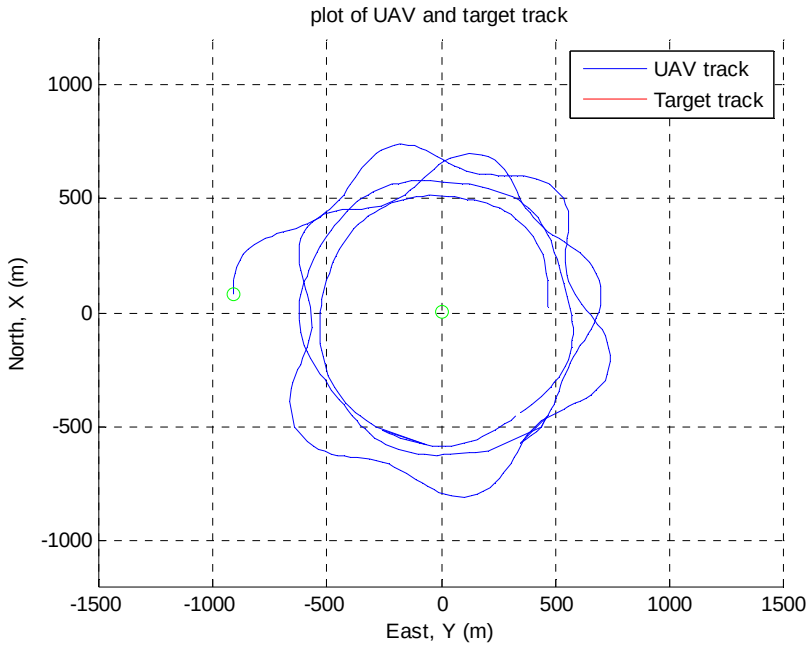


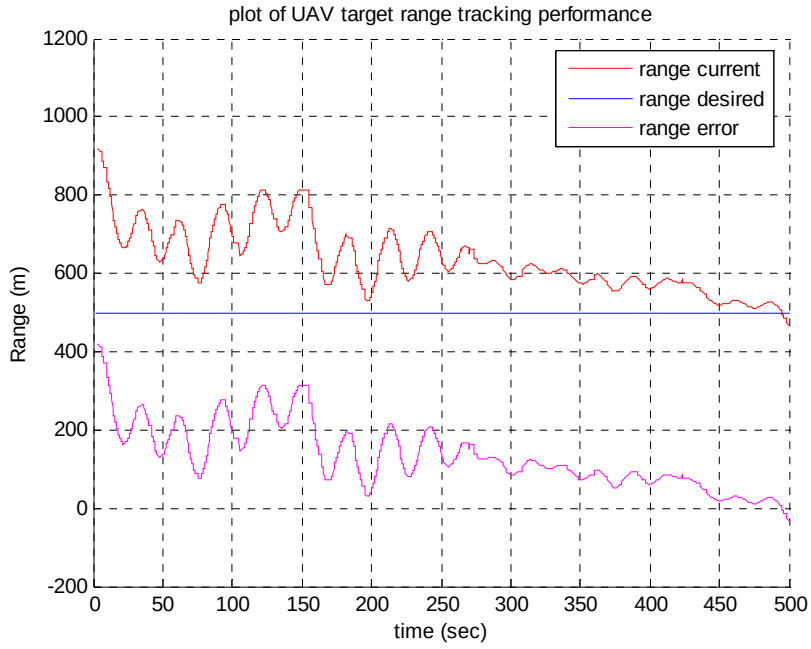
Figure 37: Sensitivity Analysis for K_1= 0.3 at Desired Range 500m (HIL)

Case 4: K_1= 0.4

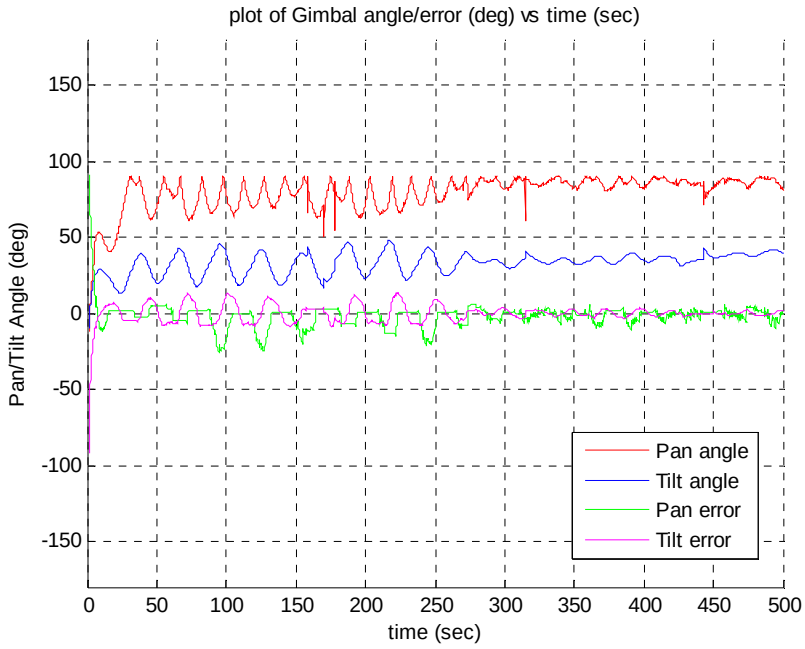
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 500m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

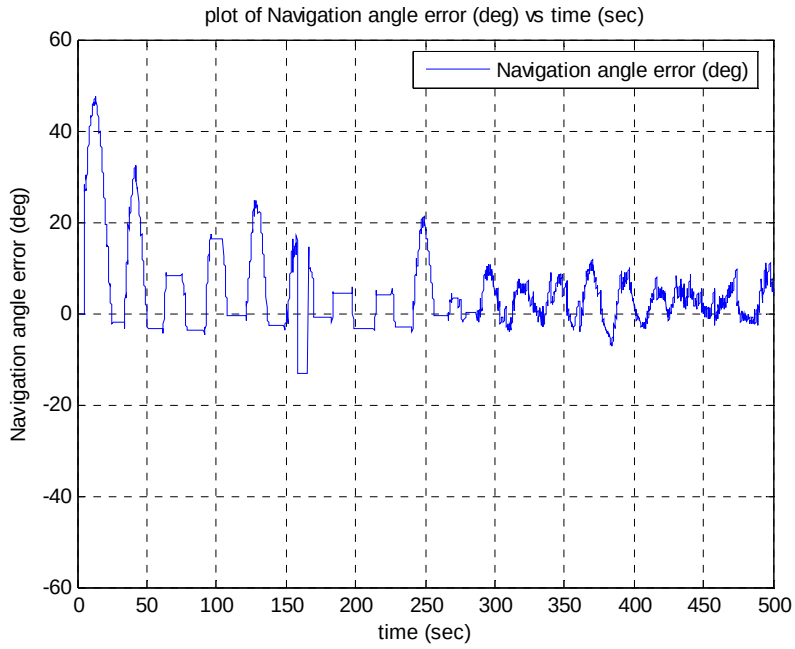


Figure 38: Sensitivity Analysis for K_1= 0.4 at Desired Range 500m (HIL)

The contributors of MOP-1 (k_1)-HIL for the four test cases are tabulated below:

K_1	Range Captured (m)	Time to first closest approach (sec)	MOP-1(k_1)-HIL, Convergence Speed (m/sec)
0.1	400	105	3.810
0.2	400	105	3.810
0.3	400	80	5.000
0.4	350	200	1.750

Table 6: MOP-1(k_1)-HIL

The MOP-1(k_1)-HIL of the UAV is plotted as follows:

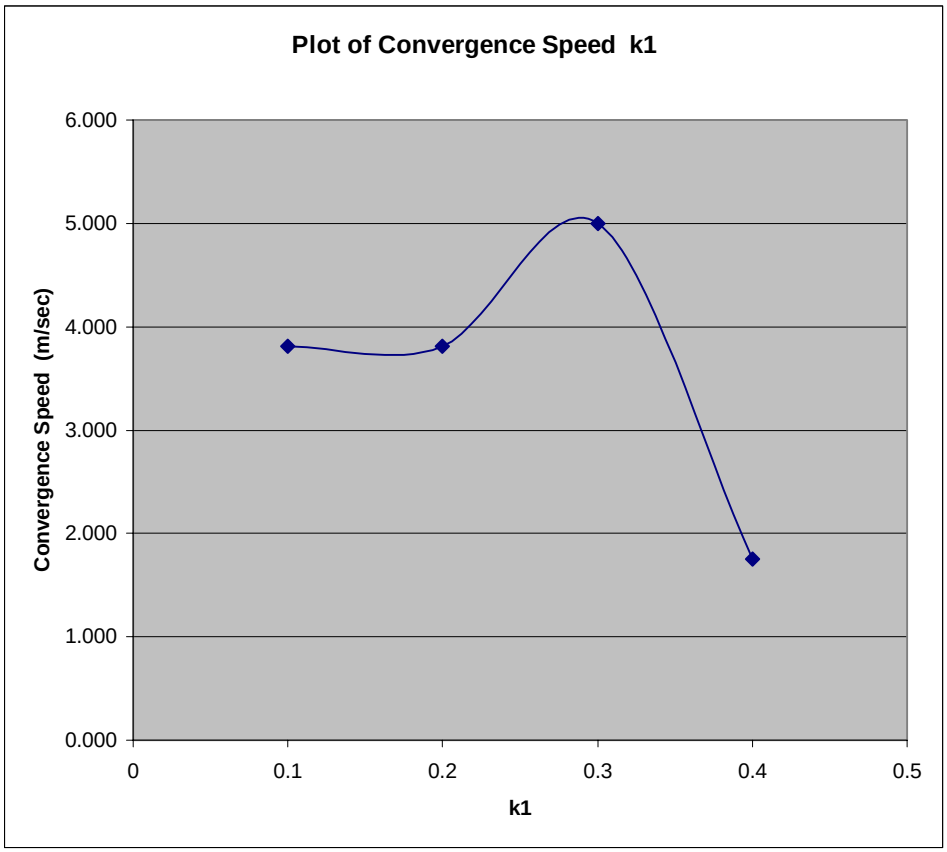


Figure 39: MOP-1(k_1)-HIL

The contributors of MOP-2 (k_1)-HIL for the four test cases are tabulated below:

K_1	Maximum range deviation error (m)	MOP-2(k_1)-HIL, Percentage deviation from desired range (%)
0.1	100	20.0
0.2	100	20.0
0.3	50	10.0
0.4	210	42.0

Table 7: MOP-2(k_1)-HIL

The MOP-2(k_1)-HIL of the UAV is plotted as follows:

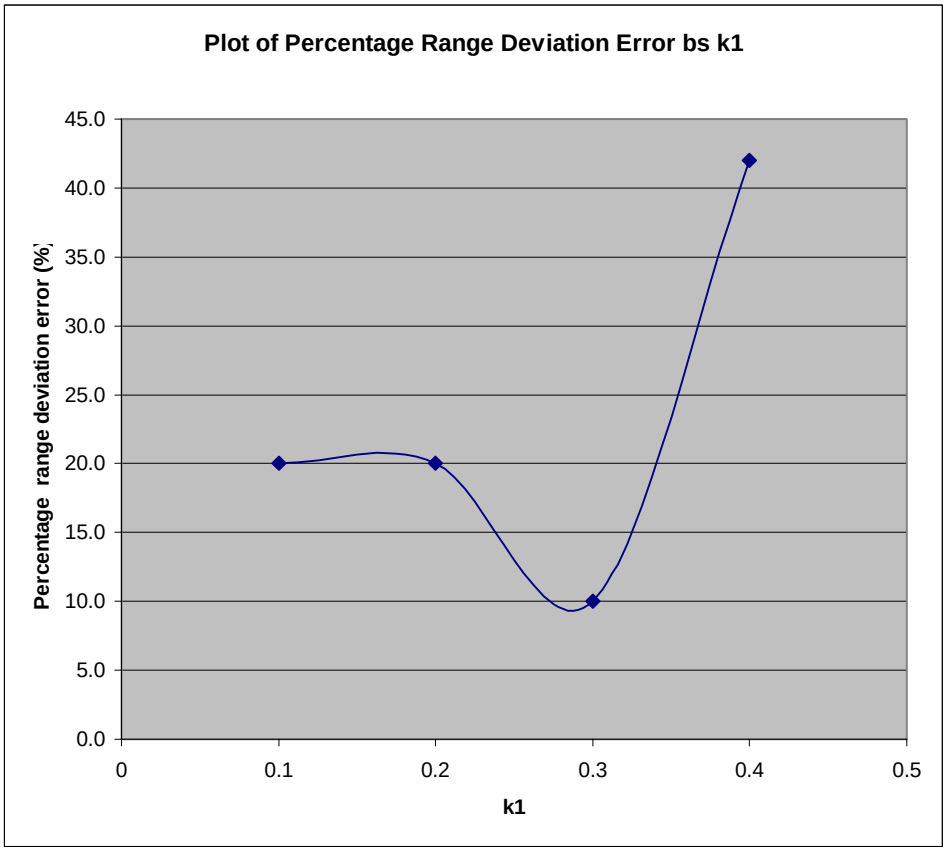


Figure 40: MOP-2(k_1)-HIL

The following observations can be made from the MOP plots and test plots above:

(a) From MOP-1(k_1)-HIL analysis, it can be observed that the convergence speed of the SUAV reaches a maximum value of 5.00 (m/sec) when k_1 is 0.3, and the lowest value of 1.750 (m/sec) when k_1 is 0.4.

(b) From MOP-2(k_1)-HIL analysis, it is observable that the percentage range deviation error of the SUAV is at the minimum at 10.0% when k_1 is 0.3; the maximum range deviation error is 42.0% when k_1 is 0.4. The range holding performance of the SUAV decreases rapidly with increasing k_1 values after the minimum value when k_1 is 0.3.

(c) The MOP (k_1)-HIL results differ with noticeable variations from SIMULINK results. However, the general SUAV convergent flight path towards the target resembles that of the SIMULINK results. The differences in the results obtained are expected, as the SUAV model and its flight characteristic in HIL are different from the SIMULINK SUAV model; moreover, the presence of hardware noises in HIL simulation also contributes to the variations.

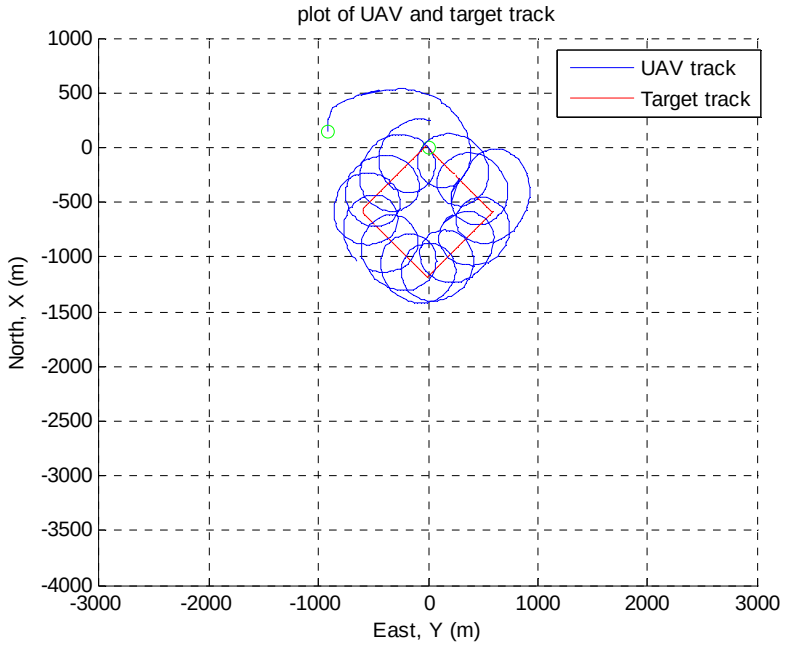
2. Sensitivity Analysis of the Control Law to Variations of *Vt/Vg*

The sensitivity analysis of the control law to variations of parameter *Vt/Vg* is examined in a scenario in which the desired range is 300m. The initial conditions are (1) SUAV velocity = 28m/s, (2) Initial position of SUAV is at [100, -900, 300], (3) Initial position of target is at [0, 0, 0] (The initial horizontal ground range from UAV to target is 900m), (4) $k_1 = 0.1$, and (5) $k_2 = 0.25$.

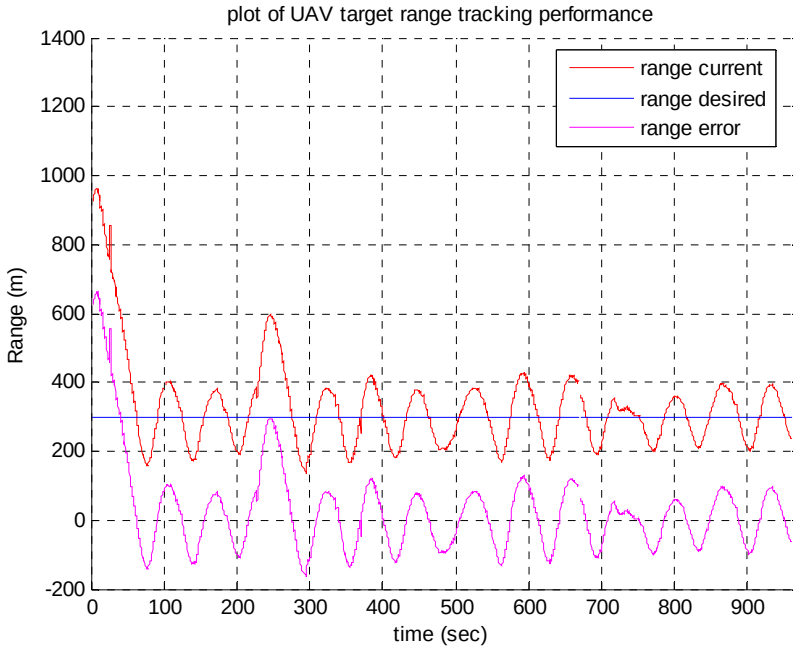
The same MOPs defined previously will be utilized to assess the performance sensitivity of the newly developed control law to parameter k_1 variations.

Case 1: Vt/Vg = 5/28

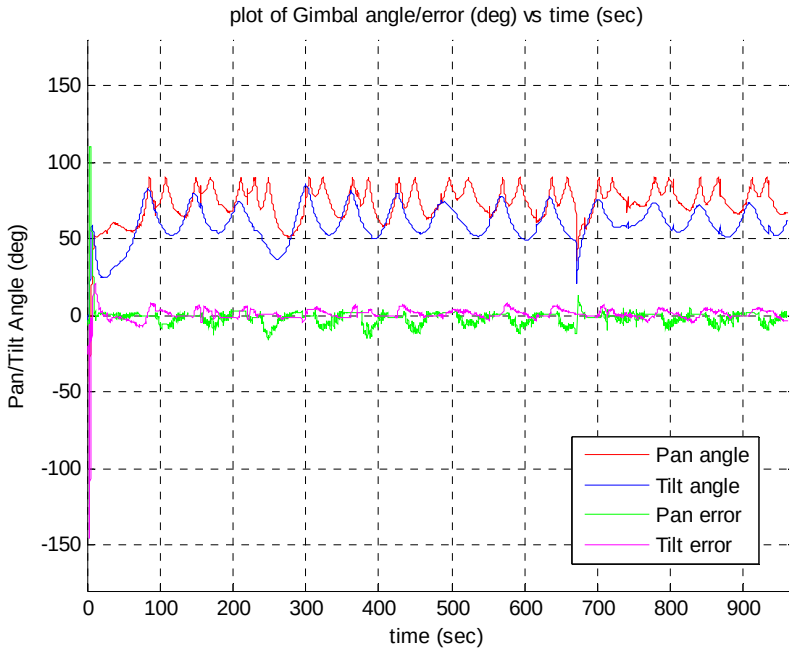
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 300m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

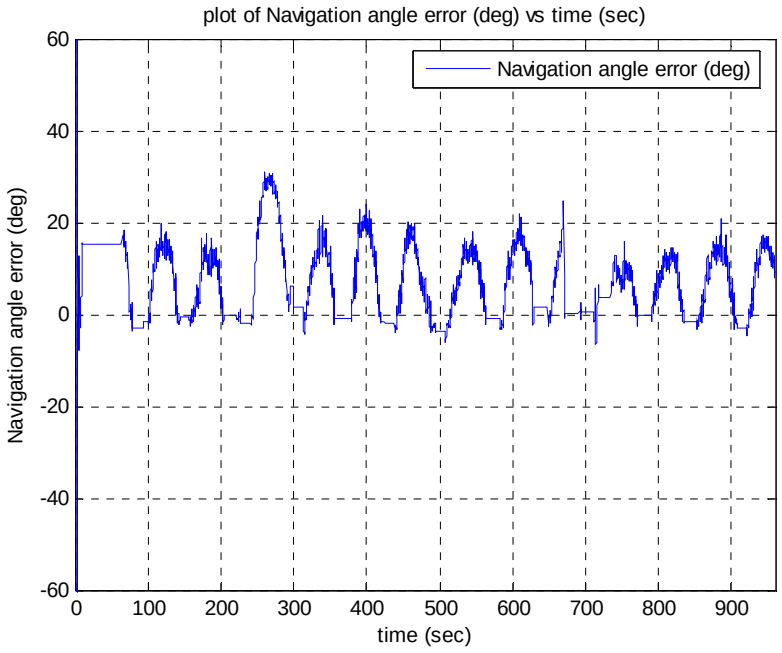
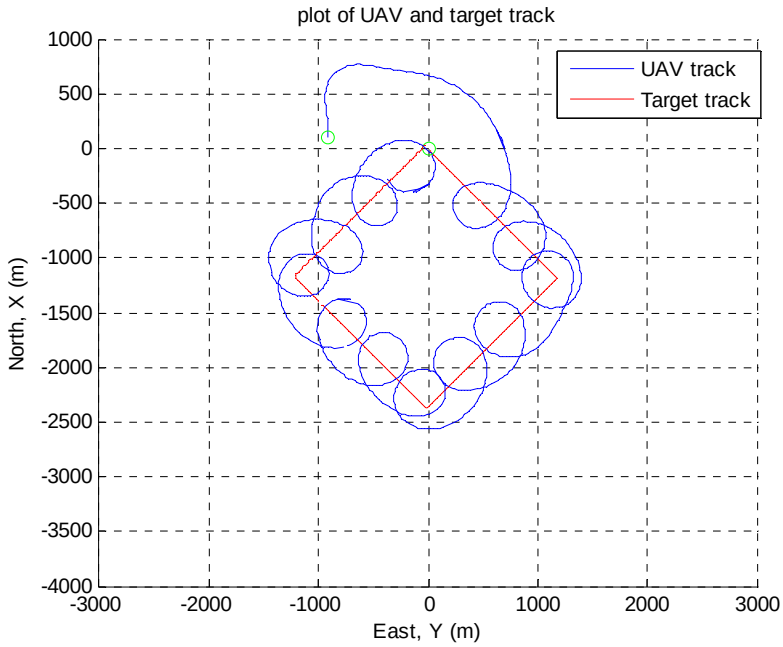


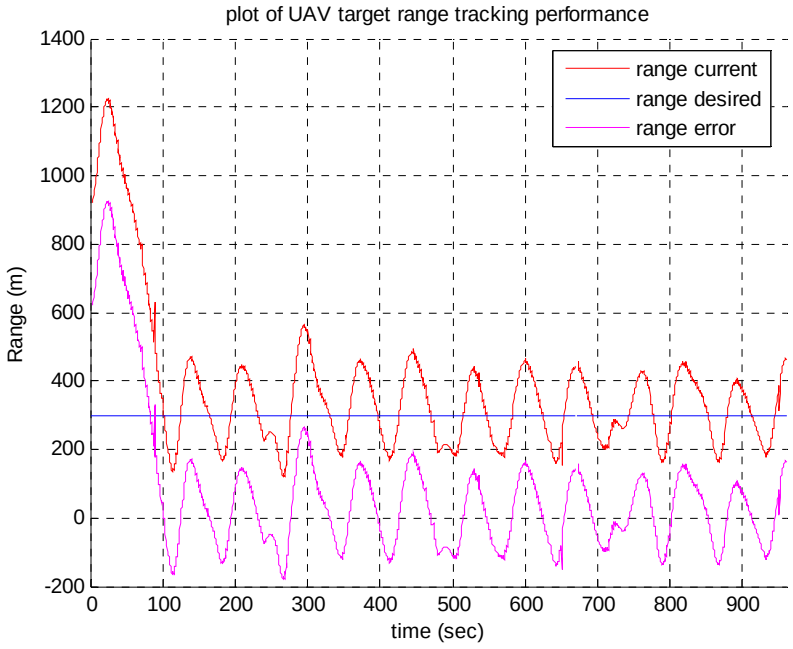
Figure 41: Sensitivity Analysis for Vt/Vg = 5/28 at Desired Range 300m (HIL)

Case 2: Vt/Vg = 10/28

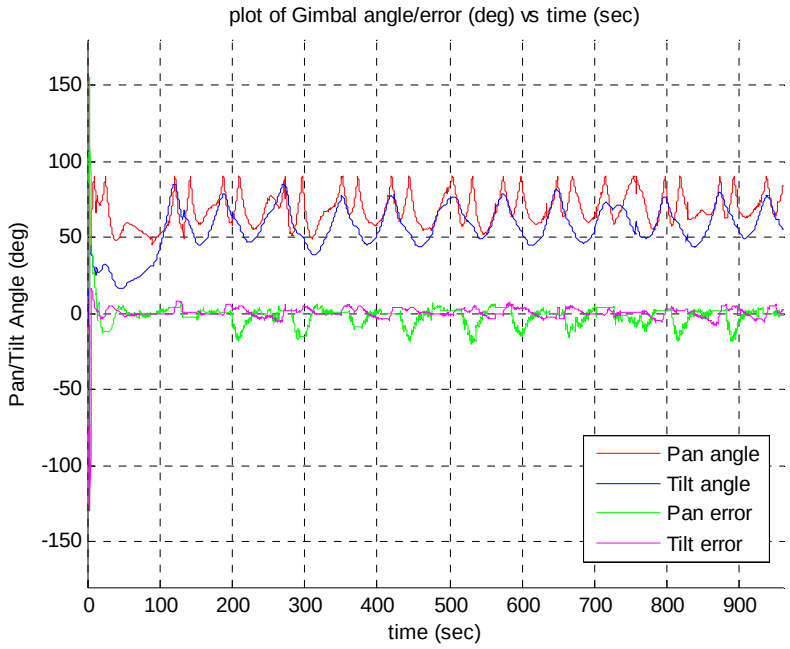
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 300m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

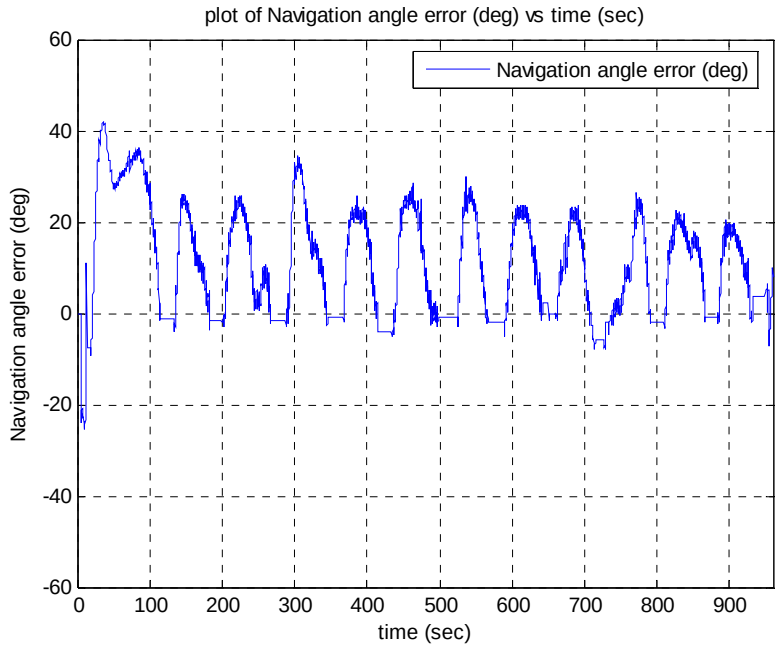
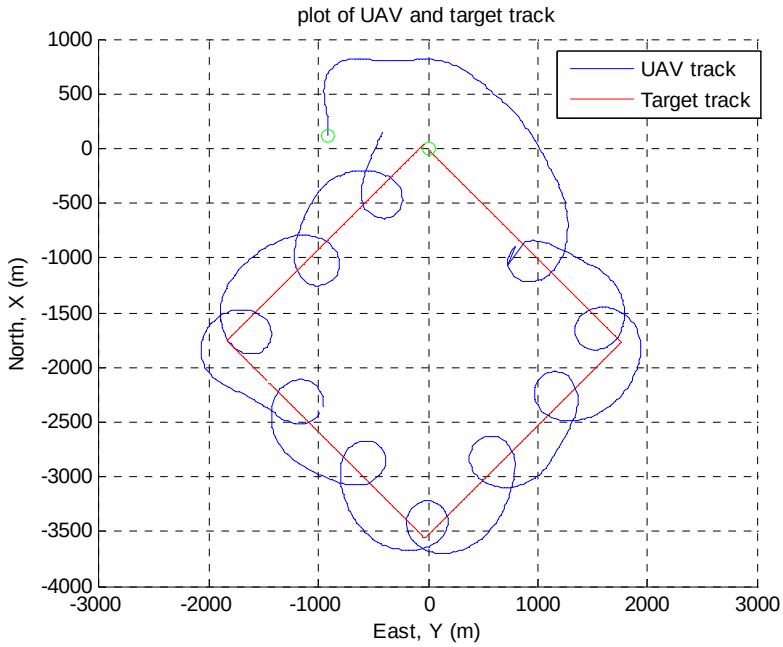


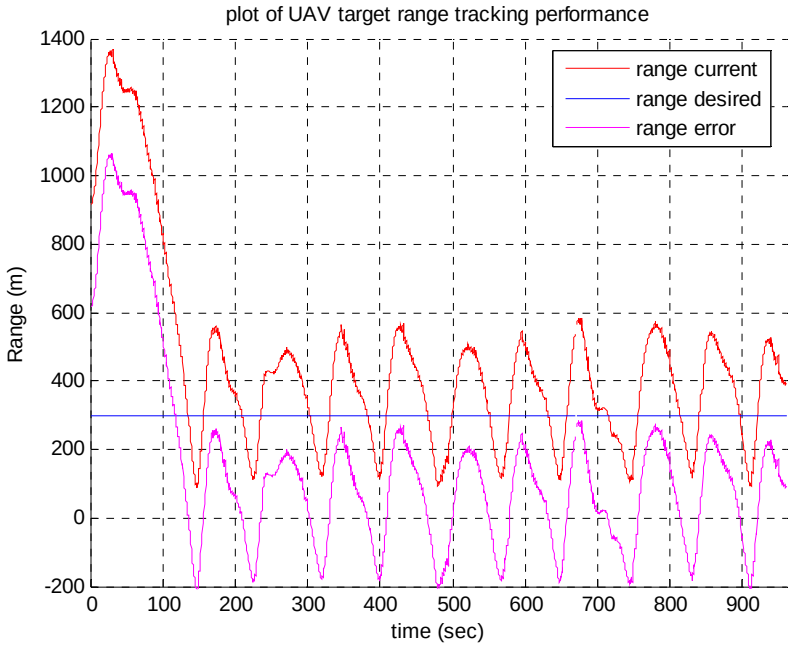
Figure 42: Sensitivity Analysis for Vt/Vg = 10/28 at Desired Range 300m (HIL)

Case 3: Vt/Vg = 15/28

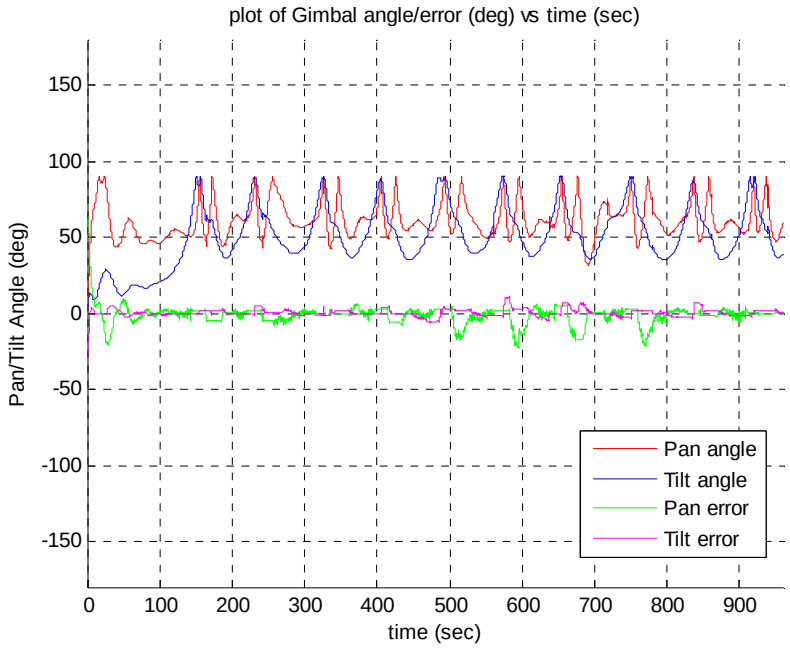
SUAV target VBTT control law tracking performance



Range convergence performance of the SUAV to the desired range of 300m



Pan / Tilt angles convergence performance and gimbal / camera angle errors



Navigation angle error convergence performance

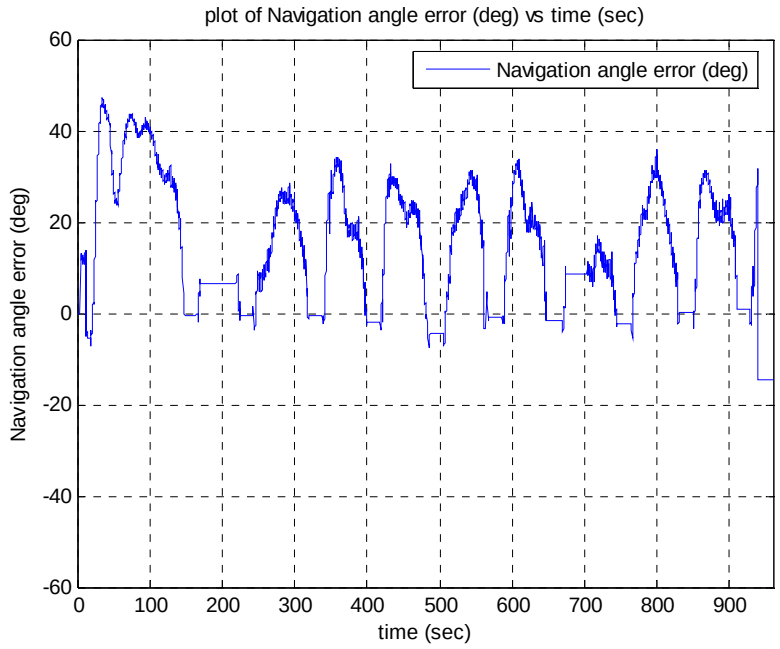


Figure 43: Sensitivity Analysis for Vt/Vg = 15/28 at Desired Range 300m (HIL)

The contributors for MOP-1 (Vt/Vg)-HIL for the three test cases are tabulated below:

Vt/Vg	Range Captured (m)	Time to first closest approach (sec)	MOP-1(Vt/Vg)-HIL, Convergence Speed (m/sec)
5/28	600	60	10.00
10/28	600	100	6.00
15/28	600	130	4.62

Table 8: MOP-1(Vt/Vg)-HIL

The MOP-1(Vt/Vg)-HIL of the UAV is plotted as follows:

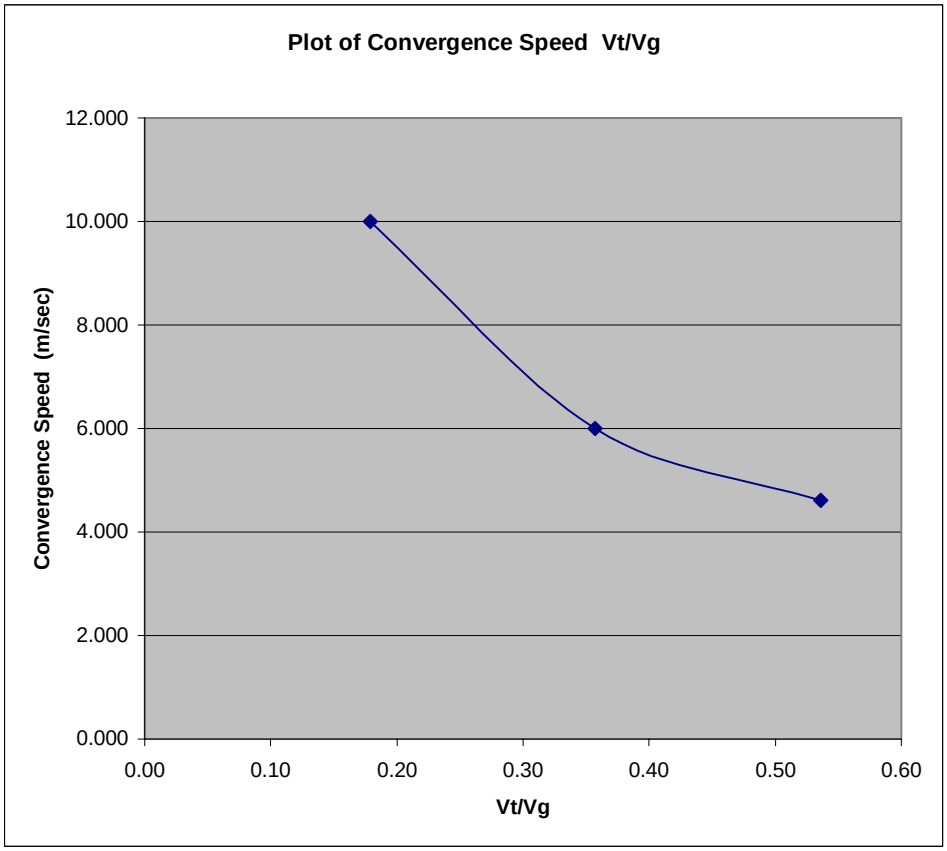


Figure 44: MOP-1(Vt/Vg)-HIL

The contributors for MOP-2 (Vt/Vg)-HIL for the three test cases are tabulated below:

Vt/Vg	Maximum range deviation error (m)	MOP-2(Vt/Vg)-HIL, Percentage deviation from desired range (%)
5/28	100	33.3
10/28	180	60.0
15/28	280	93.3

Table 9: MOP-2(Vt/Vg)-HIL

The MOP-2(Vt/Vg)-HIL of the UAV is plotted as follows:

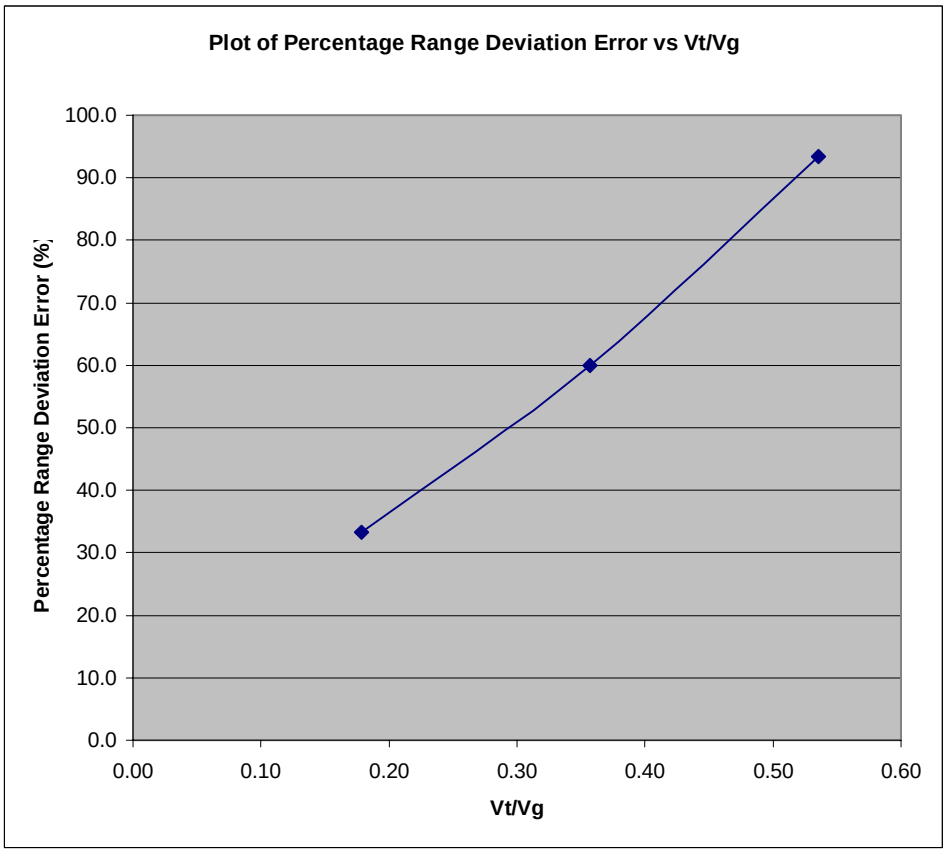


Figure 45: MOP-2(Vt/Vg)-HIL

The following observations can be made from the MOP plots and test plots above:

(a) From MOP-1(Vt/Vg) analysis, it can be observed that the convergence speed of the SUAV decreases with increasing Vt/Vg. The highest value is 10.00 (m/sec) when Vt/Vg is 5/28, and the lowest value is 4.62 (m/sec) when Vt/Vg is 15/28.

(b) From MOP-2(Vt/Vg) analysis, it is observable that the percentage range deviation error of the UAV increases with increasing Vt/Vg; the maximum range deviation error is 93.3% when Vt/Vg is 15/28, and the minimum is 33.3% when Vt/Vg is 5/28. The range holding performance of the new control law decreases rapidly with increasing Vt/Vg.

(c) The HIL simulation results are similar in trend and flight performance to their corresponding SIMULINK results.

F. HIL SIMULATION CONCLUSIONS

Although the results obtained in HIL are, in general, noisier than those of the software simulation, the HIL simulation results still exhibit similar performance. The differences in the HIL and SIMULINK results are ascribable to the inherent differences between the UAV models employed in each respective simulation. And, as stated previously, the additional hardware noises present in the HIL simulation may also be contributors to the variations in results.

Putting the slight differences and variations in the HIL and SIMULINK results aside, the HIL results obtained are similar in flight performance trend and pattern to that of the SIMULINK results. Thus the HIL results have, in essence, validated the performance and stability of the newly developed control law in the VBTT hardware.

IV. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

In comparison with the initial control law, which uses the turn bias of the UAV as a fixed quantity and does not vary with the UAV speed and desired range, the newly developed control law takes into account the UAV flight dynamics and automatically computes a dynamic turn bias commensurate with the desired range. Moreover, the camera LOS turn rate is also coupled with the UAV turn rate, thus resulting in a more efficient control of the camera. The newly developed control law is therefore more robust and efficient than the initial control law.

The results obtained from utilization of the newly developed control law in both the software simulation and the HIL simulation are encouraging and comparable to theoretical predictions. The newly developed control law is validated for the performance and stability in both the software simulation and actual hardware implementation. It lends itself readily to further flight test trials.

B. RECOMMENDATIONS

The gimbaled camera hardware should be included in the HIL simulation to test the complete robustness of the newly developed control law. The next logical goal will be to extend the simulation test to include flight trials of the new control law onboard the SUAV. At the moment of this thesis completion, preparations are underway for these tests.

During the initial convergence phase to the desired range and the final range holding at the commanded range, the VBTT control law exhibits significant oscillatory behaviors in the navigation and pan angle errors. Further improvement to the VBTT control law can be made by incorporating integral parts and/or gain scheduling techniques to the control law. These additions will allow research and examination into the possibility of damping the oscillatory behaviors in the navigation and pan angle errors for better control of the SUAV.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. Wang, I. H., Dobrokhodov, V. N., Kaminer, I. I., & Jones, K. D. (2005). "On Vision-Based Target Tracking and Range Estimation for Small UAVs." *American Institute of Aeronautics and Astronautics*. Retrieved December 05, 2005, from http://pdf.aiaa.org/preview/CDReadyMGNC05_1089/PV2005_6401.pdf.

2. Brashear, Thomas James Jr. Analysis of Dead Time and Implementation of Smith Predictor Compensation in Tracking Servo for Small Unmanned Aerial Vehicles. Master's Thesis, Naval Postgraduate School, Monterey, CA, 2005.

3. Siouris, George M. Aerospace Avionics Systems: A Modern Synthesis. New York: Academic Press, 1993.

4. Etkin, Bernard and Lloyd Duff Reid. Dynamics of Flight. Third Edition. New York: John Wiley and Sons, 1995.

5. Ogata, Katsuhiko. Modern Control Engineering. Fourth Edition. New Jersey: Prentice Hall, 2002.

6. Prince, Robert A. Autonomous Visual Tracking of Stationary Targets Using Small Unmanned Aerial Vehicles. Master's Thesis, Naval Postgraduate School, Monterey, CA, 2004.

7. Dobrokhodov, Vladimir, Issac Kaminer & Kevin D Jones, Reza Ghabcheloo (2006). "Vision Based Tracking and Motion Estimation For Moving Targets Using Small UAVs." *American Institute of Aeronautics and Astronautics*. Retrieved August 2006, from http://pdf.aiaa.org/preview/CDReadyMGNC06_1305/PV2006_6606.pdf.

8. Vagilienti, Bill and Ross Hoag. "Piccolo System User Guide." Cloud Cap Technologies, 2004, from http://www.cloudcaptech.com/ (Dec 2006).

9. "Introduction to PC104 Setup, Building XPC Model for Target PC," from http://lims.mech.northwestern.edu/students/stephens/me333/main.html (Last Access Date 4th Dec 2006).

10. "PC104 Embedded System FAQ," from http://www.controlled.com/pc104faq/ (Last Access Date 4th Dec 2006).

11. Dobrokhodov, V. N., & Lizarraga, M. (2005). "Developing Serial Communication Interfaces for Rapid Prototyping of Navigation and Control Tasks." *American Institute of Aeronautics and Astronautics*. Retrieved November 28, 2005, from http://pdf.aiaa.org/preview/CDReadyMMST05_1088/PV2005_6099.pdf.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A

MATLAB Script file for Eigenvalue stability analysis of the feedback system. (Ref 7)

```
%% Symbolic Math Toolbox script used to plot the stability analysis results
% Reduced system. UAV and gimbal controls are uncoupled
% last update on 11/14/06
clear all
clc
syms x n e r v vt dist ri DV psi real
syms k1 k2 a b c d f g positive
syms npr real
syms km1 Cr real % boundary conditions coefficients; determine size of
Omega_c

%% Introduce normalization for the state n
%  npr=n/d
x=[npr r]
%% Mapping of variables
% n = eta
% e = epsiln
% r = rho_e - =1/rho-1/rho_d
% v = Vg - UAV ground speed
% k1 - coefficient
% k2 - coefficient
% d = rho_d - desired radius
% a,b,c,f,g = constants
% dist - disturbance
% ri=1/rho

%% Nonlinear State equations including Target speed as a disturbance
%
Dnpr= (  -v*r*cos(npr*d)-k1*npr*d+vt*cos(npr*d-psi)*(r+1/d)  )/d
Dr= (r+1/d)^2*v*sin(npr*d)-vt*cos(npr*d-psi)*(r+1/d)^2 %
De= v*r*cos(npr*d)-k2*e-vt*cos(npr*d-psi)*(r+1/d)
%% Linearized state matrix
A = [diff(Dnpr,'npr') diff(Dnpr,'r') diff(Dnpr,'e');
   diff(Dr,'npr')  diff(Dr,'r') diff(Dr,'e');
   diff(De,'npr')  diff(De,'r') diff(De,'e')]
% obviously the equilibrium is at the origin
%% First iteration vt=0
%
%% Get Ao at the equilibrium for fixed target
A0 = subs(A,{npr,e,r,vt},{0,0,0,0})
% get eigenvalues of the linearized state matrix at the origin
eig(A0)
%% Analysis shows that eigenvalues are negative for all k1,k2>0
%% Second iteration vt!=0
%
%% Get Ao at the equilibrium for fixed target
A1 = subs(A,{npr,e,r,psi},{0,0,0,pi/2})
% get eigenvalues of the linearized state matrix at the origin
eig(A1)
%% Analysis shows that eigenvalues are negative for all k1,k2>0
```

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Professor Issac Kaminer
   Naval Postgraduate School
   Monterey, California

4. Professor Vladimir Dobrokhodov
   Naval Postgraduate School
   Monterey, California

5. Professor Anthony Healey
   Chairman, Department of Mechanical and Astronautical Engineering
   Naval Postgraduate School
   Monterey, California

6. Professor Yeo Tat Soon
   Director, Temasek Defence Systems Institute (TDSI)
   National University of Singapore
   Singapore